all rights reversed

HACKER

Μάης 2013 # Τιμή τεύχους 4€

τεύχος 020 - Happy Trees Edition

Πώς σας

i4G#rFq8*7rtAHh$

αποκρυπτογραφούν

το HTTPS [*]

Το ξύπνημα των μηχανών

The Invisible Internet Project

Ανάλυση πακέτων με το Wireshark

[*] ...και τι μπορείτε να κάνετε γι' αυτό

all rights reversed

HACKER

tefhos 20o

Περιεχόμενα

και οδηγίες χρήσης

Εκδότης / Διευθυντής	Χρήστος Βαρελάς
Αρχισυντάκτης	Παναγιώτης Βαρελάς
Σύμβουλοι έκδοσης	Νίκος Μουρατίδης, Βούλα Παυλίδου
Σύνταξη	Παναγιώτης Βαρελάς, Χρήστος Βαρελάς, Ανδρέας Βενιέρης, Γιώργος Γιάννου, Μανώλης Κιαγιάς, Ιωάννης Κονιάρης, Πέτρος Κυλαδίτης, Γιώργος Μελέκος, Νίκος Μουρατίδης, Νικόλαος Σηφάκης, Μέσος Παπαδόπουλος (TM), Χρίστος Τόμπρας
Γραμματική επιμέλεια	Βούλα Παυλίδου
Καλλιτεχνική Επιμέλεια	Πέτρος Φιλιππίδης

Το ηλεκτρονικό περιοδικό deltaHacker είναι συνδρομητικό και εκδίδεται από την Parabing Creations δώδεκα φορές το χρόνο, κάθε μήνα. Η Parabing Creations έχει την έδρα της στη Μαιώτιδος 3, 55133, Καλαμαριά. Ο δικτυακός τόπος του περιοδικού είναι στο http://deltahacker.gr. Το email επικοινωνίας είναι το talk2us@deltahacker.gr και ειδικά για τις συνδρομές είναι το subscriptions@deltahacker.gr. Οι γενικές πληροφορίες για τις συνδρομές παρατίθενται στο http://deltahacker.gr/subscriptions και οι ίδιες οι παραγγελίες γίνονται από το http://deltahacker.gr/pdf12order. Όλες οι απόψεις που εκφράζονται στα άρθρα δεν εκφράζουν απαραίτητα και τη γνώμη του περιοδικού.

(*) Βλέπετε τα επεισόδια του deltaCast. Δεν είναι και πολύ άσχημα.
http://deltahacker.gr/category/deltacast

Απότομη στροφή που κάνει καλό

Οι φίλοι του deltaHacker, από την εποχή που ακόμη δεν υπήρχε το περιοδικό, θα θυμούνται τις ζωηρές συζητήσεις που κάναμε το καλοκαίρι του ‘11 για τη μορφή που θα ‘πρεπε να έχει. Είκοσι μήνες μετά τον Οκτώβριο της ίδιας χρονιάς, όπου και κυκλοφόρησε το τεύχος 001 σε έντυπη μορφή, το περιοδικό άλλαξε. Μόνο που αυτή τη φορά δεν το συζητήσαμε...

by subZraw

Θυμάμαι ότι πριν καν ακόμη ξεκινήσουμε την κουβέντα για τη μορφή, έντυπη ή ηλεκτρονική, του νέου περιοδικού, ήξερα πολύ καλά ότι θα ήθελα να έχει την πρώτη. Πολλά μπορούμε να πούμε ή να μην πούμε για κάποιον που προτιμά την παραδοσιακή, έντυπη μορφή ενός περιοδικού, έναντι της λεγόμενης «ηλεκτρονικής». Για ‘μένα, που ανέκαθεν αγαπούσα το χαρτί, η επιλογή είχε να κάνει καθαρά —αν κι όχι αποκλειστικά— με τις προτιμήσεις μου. Στη συζήτηση που ξεκινήσαμε τότε, πάντως, προσπάθησα κάποια στιγμή να κάνω το συνήγορο του διαβόλου και να επιχειρηματολογήσω υπέρ ενός σεναρίου που περιελάμβανε καθαρά ηλεκτρονική μορφή, συνοδευόμενη από ένα μοντέλο επί πληρωμής πρόσβασης μέσω του `http://deltahacker.gr`. Είπαμε διάφορα, ενώ ακούστηκαν και τα επιχειρήματα όλων των πλευρών (δεν ήταν μόνο δύο). Τελικά, κάποια στιγμή φάνηκε ότι η πλειονότητα ήταν υπέρ της έντυπης μορφής. Καθόλου δεν με χάλασε αυτό, περιττό να σημειώσω.

Φαστ φόργουορντ, όπως λένε και στο καταπληκτικό χωριό μου, δύο περίπου χρόνια μετά. Το σκηνικό έχει πλέον ελαφρώς αλλάξει — κι αν τώρα δεν πληκτρολογούσα αλλά μιλούσα, η ειρωνεία σ’ αυτό το «ελαφρώς» θα ήταν ξεκάθαρη. Επιτρέψτε όμως μου να εξηγηθώ. Κοιτάζοντας με συνεργάτες το αναγνωστικό κοινό του περιοδικού και συγκρίνοντάς το μ’ αυτό που ήταν δώδεκα ή ακόμη κι έξι μήνες πριν, με χαρά βλέπουμε ότι διευρύνεται σταθερά και γρήγορα. Την ίδια στιγμή, όμως, γύρω μας συμβαίνουν ένα σωρό αλλαγές. Μιλάω για εκείνες τις αλλαγές οι οποίες —θα συμφωνήσετε φαντάζομαι κι εσείς—, για τον περισσότερο κόσμο δεν είναι και τόσο ευχάριστες. Δεν νομίζω ότι χρειάζεται να πούμε κάτι επ’ αυτού. Στο κάτω κάτω, όλοι στον ίδιο υποχώρο του χωροχρονικού συνεχούς κινούμαστε και δραστηριοποιούμαστε, οπότε λίγο πολύ όλοι τις λουζόμαστε.

Πώς όμως επηρεάζουν το αγαπημένο μας περιοδικό αυτές οι καταιγιστικές, όχι-και-τόσο-ευχάριστες αλλαγές; Θα σας πω αμέσως, πώς το επηρεάζουν. Εδώ και πολλούς μήνες, που λέτε, σχεδόν από το φθινόπωρο του ‘12, οι συνδρομητές του περιοδικού οι οποίοι μπορούν να στηρίξουν την έντυπη μορφή του, δηλαδή εκείνοι με τουλάχιστον 6μηνη συνδρομή, έχουν λιγοστέψει επικίνδυνα. Βλέπετε, ολοένα και περισσότεροι φτάνουν στο σημείο, εκεί, γύρω στις 20 του μήνα, όπου με τα χρήματα που τους περισσεύουν —αν τους περισσεύουν— έχουν να πάρουν αποφάσεις του στιλ «να πληρώσω το κινητό και να πάω για κανέναν καφέ ή ν’ ανανεώσω τη συνδρομή μου στο δέλτα-χάκερ» [*].

Αφού πάρουν την απόφασή τους, μη νομίζετε, δεν μας γυρίζουν την πλάτη. Αντίθετα, έρχονται οι άνθρωποι και με διάφορους τρόπους μάς ζητούν ουσιαστικά το ίδιο πράγμα: Να μειώσουμε τις τιμές. Από τη μεριά μας, έχουμε προσπαθήσει επανειλημμένως να μειώσουμε το κόστος παραγωγής και διανομής. Δεν λέω, κάτι καταφέραμε, αλλά να συμπιέσουμε περισσότερο το συνολικό κόστος πλέον δεν μπορούμε. Κάναμε λοιπόν προσφορές με εκπτώσεις, μετά τις επαναλάβαμε με ακόμη μεγαλύτερες. Δεν είναι ότι δεν βρήκαν ανταπόκριση, το πρόβλημα όμως παρέμενε: Κατά πώς φαινόταν, το κόστος το περιοδικού έπρεπε να πέσει κι άλλο — και μάλιστα δραματικά.



Και κάπως έτσι μάς προέκυψε η ηλεκτρονική έκδοση του deltaHacker — συγκεκριμένα σε μορφή PDF. Το τεύχος 019 ήταν το τελευταίο που βγήκε σε έντυπη μορφή, ενώ από το 020 το περιοδικό είναι αποκλειστικά συνδρομητικό. Να σας πω την αλήθεια, από το τεύχος 018 έπρεπε να έχουμε κάνει την αλλαγή. Πληρώσαμε ακριβά τη διστακτικότητά μας, αλλά δεν πειράζει. Αυτό που είδαμε με τις νέες τιμές που έφερε η ηλεκτρονική έκδοση (`http://deltahacker.gr/pdf12order`), ήταν μια μεγάλη κι απότομη αύξηση στις ανανεώσεις συνδρομών. Ταυτόχρονα, πολλοί νέοι αναγνώστες ήλθαν στο deltaHacker — και μιλάμε για ανθρώπους που δεν είχαν πάρει ούτε ένα έντυπο τεύχος! Το πιο σημαντικό, όμως, είναι ότι κι από τους νέους μας φίλους συνεχίζουμε ν’ ακούμε καλά λόγια για τη δουλειά μας! Είναι νομίζω περιττό να πω πόσο χαιρόμαστε γι’ αυτό :D

Η επιλογή του PDF format για την ηλεκτρονική έκδοση του deltaHacker, δεν έγινε τυχαία. Βασικά, ήταν που θέλαμε μια ποιοτική αλλά και ταυτόχρονα φορητή μορφή, που να «παίζει» παντού: σε tablets, smartphones, phablets και φυσικά σε κάθε υπολογιστή, ανεξαρτήτως του λειτουργικού που τρέχει. Ένα άλλο καλό με τα PDF μας, είναι η δυνατότητα για ελεύθερη αναζήτηση κειμένου. Ήδη έχουμε πολλά θετικά σχόλια γι’ αυτό το χαρακτηριστικό, αφού επιτρέπει τον ταχύτατο εντοπισμό θεμάτων που έχουν δημοσιευτεί σε προηγούμενα τεύχη. (Σε μορφή PDF διατίθενται κι όλα τα προηγούμενα τεύχη, ξεκινώντας από το 001.) Αλλά τα «συν» δεν σταματούν εδώ. Από τα PDF των τευχών, μπορείτε να επιλέγετε κείμενο και να κάνετε copy-paste. Ιδιαίτερα βολικό αυτό, ειδικά όταν θέλετε να αντιγράψετε σε κάποιον editor τμήματα κώδικα. Επίσης, ένα άλλο καλό είναι ότι επιτρέπεται η εκτύπωση. Αν λοιπόν κάποιος βολεύεται να διαβάσει ένα ή περισσότερα άρθρα σε χαρτί, τα PDF μας δεν έχουν κλειδωμένη τη σχετική δυνατότητα.

Αστεία αστεία, πάντως, με τη στροφή στην ηλεκτρονική έκδοση είδαμε ότι ο κόσμος που έχει και χρησιμοποιεί tablet είναι *πολύς* περισσότερος απ’ όσο εκτιμούσαμε. Προσωπικά, παραδέχομαι ότι μαύρα μεσάνυχτα είχα επί του θέματος. Χαίρομαι βέβαια που η πραγματικότητα δεν είναι όπως τη φανταζόμουν. Πολύ περισσότερο όμως χαίρομαι που το περιοδικό φτάνει, πλέον, σε περισσότερο κόσμο — κι αφήνει θετικότατες εντυπώσεις.

Είναι κι άλλα που θα ήθελα να σας πω, γύρω από συγκεκριμένες αποφάσεις που πήραμε. Βάζω όμως προς το παρόν μια άνω τελεία, μιας και είναι επιτέλους ώρα να κυκλοφορήσουμε το νέο τεύχος. Μας άλλαξαν λίγο τα φώτα τα logistics της μετάβασης —αν μη τι άλλο είχαμε να επεκτείνουμε και τις συνδρομές εκατοντάδων αναγνωστών—, αλλά φυσικά τα φέραμε βόλτα. Θα συνεχίσουμε την κουβέντα μας κάποια άλλη στιγμή. Προς το παρόν, σας αφήνω να απολαύσετε το ολοκαίνουργιο τεύχος!

***[*]** Φήμες λένε ότι κάποιοι το προφέρουν «ντέλτα-χάκερ». Καλά κάνουν, ιφ γιου ασκ μι.*

POST CAR

THE ADDRESS TO BE WRITTEN ON THIS SI



ονέιλ, μέιλ -

Αγαπητοί μου,

Το παραδέχομαι. Κάθε φορά κάνω το ίδιο λάθος. Λέω στον εαυτό μου, αυτό τον μήνα δεν θα μιλήσεις για πολιτικά, τέρμα. Οι άνθρωποι έχουν πάρει το περιοδικό τους και θέλουν να διαβάσουν τα δικά τους. Το τελευταίο που θέλουν είναι εσένα και τα κηρύγματά σου.

Το λέω και το ξαναλέω για να το πιστέψω. Και τελικά το πιστεύω.

Κάθομαι, λοιπόν, στη γραφομηχανή, βάζω τη λευκή σελίδα και ξεκινώ με όλες τις καλές προθέσεις.

Και ναι: Σε κάθε μου επιστολή, οι πρώτες δυό τρεις παράγραφοι έχουν πράγματι να κάνουν με τα τρέχοντα. Πότε η Καθαρή Δευτέρα και η Σαρακοστή, πότε το χριστουγεννιάτικο δέντρο, πότε η θέα από το περιστρεφόμενο εστιατόριο στο Κόρυο, πότε το Πόρτο Γερμενό και τα τζιτζίκια.

Μετά όμως κάτι συμβαίνει και παρασύρομαι. Νομοτελειακά, λες.

Την περασμένη φορά, ας πούμε, ένας Θεός ξέρει πώς από την Ελληνική Σαρακοστή πήγα στο Βρετανικό Πάσχα, κι από εκεί, μέσω των λιοκαμένων πλην γαλανόλευκων βλασταριών, στους αρχαίους ημών προγόνους και τους συγχρόνους ημών ψεκασμένους.

Νομίζετε ότι είχα σκοπό να μιλήσω για όλα αυτά; Όχι βέβαια.

Εγώ, άλλα πράματα με απασχολούν από καιρό, γι' αυτά θα ήθελα να μιλήσω. Αλλά πού ν' αγιάσω; Νομίζετε μ' αφήνουν;

Με το που κάθομαι στη γραφομηχανή, λες και τους κάνω σινιάλο, πλακώνουν όλοι από δίπλα και αρχινάνε τα δικά τους. Χαμός. Είμαι σίγουρος ότι το κάνουν επίτηδες. Δεν ξέρω, μην έχουν βάλει και κανένα στοίχημα με τον ανιψιό. Ο οποίος ανιψιός, παρεμπιπτόντως, έχει έρθει αυτές τις μέρες από τη Γηραιά Αλβιώνα (Πάσχα, γαρ) και μας το παίζει ψύχραιμος παρόλο τον κάζο με τα μπιτκόιν που αγοραζοπούλαγε τις προάλλες.

Τέλος πάντων, δεν θα υπεισέλθω σε λεπτομέρειες· το θέμα είναι ότι ακόμα και σήμερα, κοτζάμ Κυριακή του Πάσχα στη βεράντα της Κυρίας Καίτης στο Πόρτο Γερμενό, με το που έκατσα στη γραφομηχανή κάτω από τα πεύκα, να 'σου τους από δίπλα με τα πασχαλινά τους τα αβγά, τις εφημερίδες τους και τα γιουτιούμπ τους.

Μόνο η μητέρα της εξαδέλφης, που είναι Ινδουίστρια, είναι μέσα και ασχολείται με κάτι δημιουργικό. Ψήνει το κοκορέτσι. Όλοι οι άλλοι είναι εδώ και με ταλαιπωρούν.

Να πεις ότι αυτό γίνεται μόνο σήμερα, να το καταλάβω. Αλλά όχι. Δεν περιμένανε το Πάσχα. Κάθε μέρα τα ίδια. Χαΐρι δεν έχω δει.

Και τι δεν έχω ακούσει, αγαπητοί μου. Τα πάντα. Είχα αποφύγει αυτές τις μέρες τις ειδήσεις, αλλά τώρα τα μαθαίνω όλα.

Ο ένας παίρνει, λέει, τα κουμπούρια και αρχινάει να βαράει στο ψαχνό τους κακομοίρηδες που του μαζεύουν τις φράουλες, επειδή του ζητήσανε τα μεροκάματά τους. Στο ψαχνό για εκφοβισμό, λέει. Νόμιζε ότι τον απειλούσαν. Εμ, βέβαια, να τους εκφοβίσει. Πιο εκφοβισμένος από τον ψόφιο,

γίνεται; Δεν γίνεται. Γκουντ.

Ο άλλος θα φτιάξει σάιτ για την ενημέρωση των πολιτών του Δήμου Ιεράς Πόλεως Μεσολογγίου σχετικά με δημοτικές δραστηριότητες, και θα πληρώσει (όχι αυτός βέβαια) 308.509,66 ευρώ.

Ιερά η πόλις, οκέι; Ιεροτάτη. Αλλιώς δεν εξηγείται. Ούτε το ποσό, ούτε ο επαγγελματισμός του εκτιμητή του έργου που το κοστολόγησε. Να το ξαναπώ, να το εμπεδώσουμε. Τριακόσιες οχτώ χιλιάδες πεντακόσια εννέα ευρά, για το σάιτ του Δήμου Ιεράς Πόλεως Μεσολογγίου. Και εξήντα έξη σεντς.

Πάω να τους το φτιάξω εγώ το σάιτ. Φεύγω τώρα.

Τέλος πάντων. Δεν πρέπει να ταράζομαι. Μικρό διάλειμμα, να πάρω την ολανζαπίνη. Ο γιατρός μού την έχει γράψει PRN.

Άντε τώρα να γράψω. Τι να γράψω; Τι άλλο να γράψω; Εδώ οι άνθρωποι το έχουν τερματίσει. Το μυαλό θολώνει. Μπλοκάρει.

Θα μου πείτε, το συγγραφικό μπλοκάρισμα είναι ένδειξη ελευθερίας. Αντί να χαίρομαι που μου έχει δώσει το ελεύθερο ο Κύριος Εκδότης (ΤΜ) να γράφω ό,τι θέλω, εγώ κάθομαι και παραπονιέμαι.

Μα, όχι, δεν παραπονιέμαι καθόλου για αυτό. Δεν με ενοχλεί η ελευθερία. Περισσότερο με ενοχλεί που, κοτζάμ ελευθερία, εγώ καταλήγω να γράφω ξανά και ξανά για τα ίδια, για τις τριακόσιες οχτώ χιλιοπούλες του σάιτ της Ιεράς Πόλεως, για τους τυφλούς της Ζακύνθου, για τους χρυσούς αβγούς και τα συσσίτια για Έλληνες (ΤΜ), για τους ψεκασμούς, για το ελληνοπρεπές μας τίποτα.

Ξέρετε, το ζήτημα με έχει απασχολήσει. Το ζήτημα της ελευθερίας, εννοώ.

Τελικά υπάρχει ελευθερία ή είναι μύθος; Αν πράγματι είμαι ελεύθερος να γράφω ό,τι μου γουστάρει, τότε γιατί καταλήγω --νομοτελειακά-- να γράφω τα ίδια;



Θα μου πείτε, /αυτό/ σημαίνει ελευθερία, να μπορείς να γράφεις τα ίδια, αν και εφόσον αυτό γουστάρεις.

Ναι, αλλά αναρωτιέμαι μήπως αυτό που ονομάζουμε ελευθερία είναι μόνο κατ' όνομα ελευθερία. Μήπως στην πραγματικότητα το μόνο που μπορούμε να κάνουμε είναι να ακολουθούμε πιστά ένα κρυφό σενάριο, που έχει ήδη γραφτεί και το οποίο μας αποκαλύπτεται καθώς το ακολουθούμε;

Αγαπητοί μου.

Οι παλαιότεροι από εσάς θα αναγνωρίζετε στα ανωτέρω ερωτήματα τους προβληματισμούς που έχω επανειλημμένως εκθέσει σε άλλες επιστολές μου.

Θα το έχετε καταλάβει, είμαι σίγουρος, ότι τα ερωτήματα είναι στην πραγματικότητα ρητορικά. Ως αφορμή τα εκλαμβάνω, για να μιλήσω γι' αυτό που με απασχολεί εδώ και καιρό: Με απασχολεί το λογικό σφάλμα της σύγχυσης μεταξύ μιας οντότητας και της περιγραφής μιας οντότητας.

Σας τα έχω πει και τα έχω ξαναπεί. Αλλά ας το δούμε εδώ επί του συγκεκριμένου. Ας κάνουμε ένα νοητικό πείραμα.

Πείτε, λοιπόν, ότι τελικά δεν υπάρχει ελευθερία, και ότι όλες σας οι ενέργειες είναι καθοριζόμενες νομοτελειακά. Φανταστείτε, επιπλέον, πως μια θεότης σάς παρακολουθεί και γνωρίζει τις λεπτομέρειες αυτής της νομοτέλειας. Ξέρει τι πρόκειται να κάνετε πριν καν εσείς ξεκινήσετε να το σκέφτεστε.

Φανταστείτε, τώρα, ότι ετοιμάζεστε να παίξετε Λόττο, και πιάνετε το μολύβι για να συμπληρώσετε τους αριθμούς. Πετάγεται, λοιπόν, η θεότης από δίπλα και σας λέει: «Νομίζεις ότι είσαι ελεύθερος να διαλέξεις όποιον αριθμό θέλεις, αλλά εγώ ξέρω ποιους θα διαλέξεις. Θα διαλέξεις το 3, το 14, το 15, το 9, το 2, και το 6.»

Τι θα κάνατε αν σας συνέβαινε κάτι τέτοιο;

Για μένα, ξέρω. Θα τσέκαρα να δω μήπως έχει λήξει η ολανζαπίνη.

Σοβαρά τώρα, τι θα κάνατε;

Λοιπόν, αν εμένα η θεότητα μου το έλεγε ότι θα παίξω αυτούς τους τάδε αριθμούς, από αντίδραση θα έπαιζα άλλους. Ακόμα και αν συνειδητοποιούσα ότι η θεότητα έδειξε πράγματι να ξέρει πριν από μένα για μένα.

Ιδίως τότε.

Γεμάτος θαυμασμό θα της έβγαζα το καπέλο (της θεότητας) και θα έπαιζα άλλους αριθμούς, π.χ., τους 2, 7, 1, 8, 28, 18.

Θα μου πείτε, σιγά, η θεότητα θα το είχε προβλέψει και αυτό.

Ναι, σωστά, αλλά μην μπερδευόμαστε. Το τι θα έχει προβλέψει η θεότητα και τι όχι, δεν είναι το σημαντικό. Το σημαντικό είναι ότι κατά τη στιγμή ακριβώς που είναι να συμπληρώσω αριθμούς στο δελτίο του Λόττο, τους αριθμούς τους συμπληρώνω /εγώ/, όχι η θεότητα.

Με άλλα λόγια: Στο επίπεδο της πράξης, είμαι πάντα ελεύθερος.

Βεβαίως, στο «μετα»-επίπεδο της περιγραφής-ερμηνείας της πράξης, μπορεί και να μην είμαι. Αλλά δεν έχει σημασία. Σημασία έχει ότι δεν δύναμαι /ποτέ/ να βρίσκομαι στα δύο αυτά επίπεδα /ταυτόχρονα/. Βρίσκομαι είτε στο ένα, είτε στο άλλο. That's it.

Διακόπτω γιατί ετοιμαζόμαστε να τσουγκρίσουμε τα αβγά.

Σκεφτείτε το και θα επανέλθω.

Σας ασπάζομαι,
Θείος Ακάκιος

openSUSE Conference

18-22 July 2013

Thessaloniki, Greece

Olympic Museum

See you there !!!

conference.opensuse.org

#opensuse-conference @ Freenode



Skill: Beginner

Tags: TLS, SSL, HTTPS, certificate, Certificate Authority, CA, Man-in-The-Middle, MiTM, proxy, transparent proxying, iptables, pf

Πώς σας i4G#rFq8*7rtAHh$ αποκρυπτογραφούν το HTTPS [*]

...κι αναφερόμαστε φυσικά στους λάθος ανθρώπους. Γιατί, πράγματι, όσο ασφαλές κι αν είναι το HTTPS, τόσο εύκολα μπορεί να αποκρυπτογραφηθεί από κάποιον τρίτο – αρκεί βέβαια να πληρούνται ορισμένες προϋποθέσεις. Στο παρόν άρθρο συζητάμε γι' αυτές, ενώ δεν αποφεύγουμε να κάνουμε και τις σχετικές δοκιμές αποκρυπτογράφησης ;)

by sub2raw

Το TLS (Transport Layer Security) είναι ένα πρωτόκολλο που έχει σχεδιαστεί για την εγκαθίδρυση ασφαλών, κρυπτογραφημένων καναλιών επικοινωνίας στο Internet. Συχνότερα απ' όσο πρέπει γίνεται αναφορά στο SSL (Secure Sockets Layer), το οποίο στην πραγματικότητα είναι ο προκάτοχος του TLS. Σημασία πάντως έχει ότι το TLS/SSL χρησιμοποιείται κατά κόρον για να παρέχει την επιθυμητή ασφάλεια σε πληθώρα εφαρμογών, όπως, π.χ., ηλεκτρονική αλληλογραφία, instant messaging και VoIP. Επιστρατεύεται βεβαίως και στο web browsing, όπου τότε μιλάμε για το πρωτόκολλο HTTP Secure ή απλά HTTPS. Ακριβέστερα, αυτό που γίνεται είναι ότι το δημοφιλέστατο πρωτόκολλο HTTP (HyperText Transfer Protocol) υλοποιείται «πάνω» από το TLS/SSL. Έτσι, όταν η σύνδεση προς κάποιον web server είναι HTTPS, τότε τα διακινούμενα δεδομένα είναι κρυπτογραφημένα και προστατεύονται από τεχνικές όπως, π.χ., το packet sniffing, υπό την έννοια ότι δεν εκτίθενται σε τρίτους. Κάθε φορά λοιπόν που συνδεόμαστε μέσω HTTPS στο Facebook ή στην τράπεζά μας, οι μόνοι που μπορούν και αποκρυπτογραφούν τα διακινούμενα δεδομένα είναι ο web browser κι ο web server, δηλαδή μόνο τα δύο άκρα του καναλιού επικοινωνίας. Αλλά αυτό είναι κάτι που, προφανώς, το θέλουμε.

Ένα από τα καλά του HTTPS είναι ότι κάνει δυνατή και την πιστοποίηση (authentication) των απομακρυσμένων web servers. Πράγματι, όποτε επισκέπτομαι την τράπεζά μου ή το Facebook μέσω HTTPS, διαθέτω ένα άριστο εργαλείο για να βεβαιώνομαι ότι πράγματι συνδέθηκα στην τράπεζα ή στο Facebook, κι όχι σε κάποιο site-κλώνο που έχει στήσει ένας απατεώνας. Με άλλα λόγια, το HTTPS παρέχει προστασία κατά των επιθέσεων Man-in-The-Middle (MiTM attacks).

Στις επιθέσεις του είδους αναφερόμαστε συχνά στο περιοδικό — κι αυτή τη στιγμή στο μυαλό του γράφοντα έρχονται δύο σχετικά άρθρα που φιλοξενούνται στο deltaHacker 001 (βλ. http://deltahacker.gr/?p=2041 και http://deltahacker.gr/?p=2037). Το θέμα όμως με τις επιθέσεις MiTM είναι ότι υλοποιούνται μ' ένα σωρό τρόπους. Υπό προϋποθέσεις, μάλιστα, ένας εξ αυτών είναι ικανός να ακυρώσει την όλη ασφάλεια που προσφέρει το HTTPS!

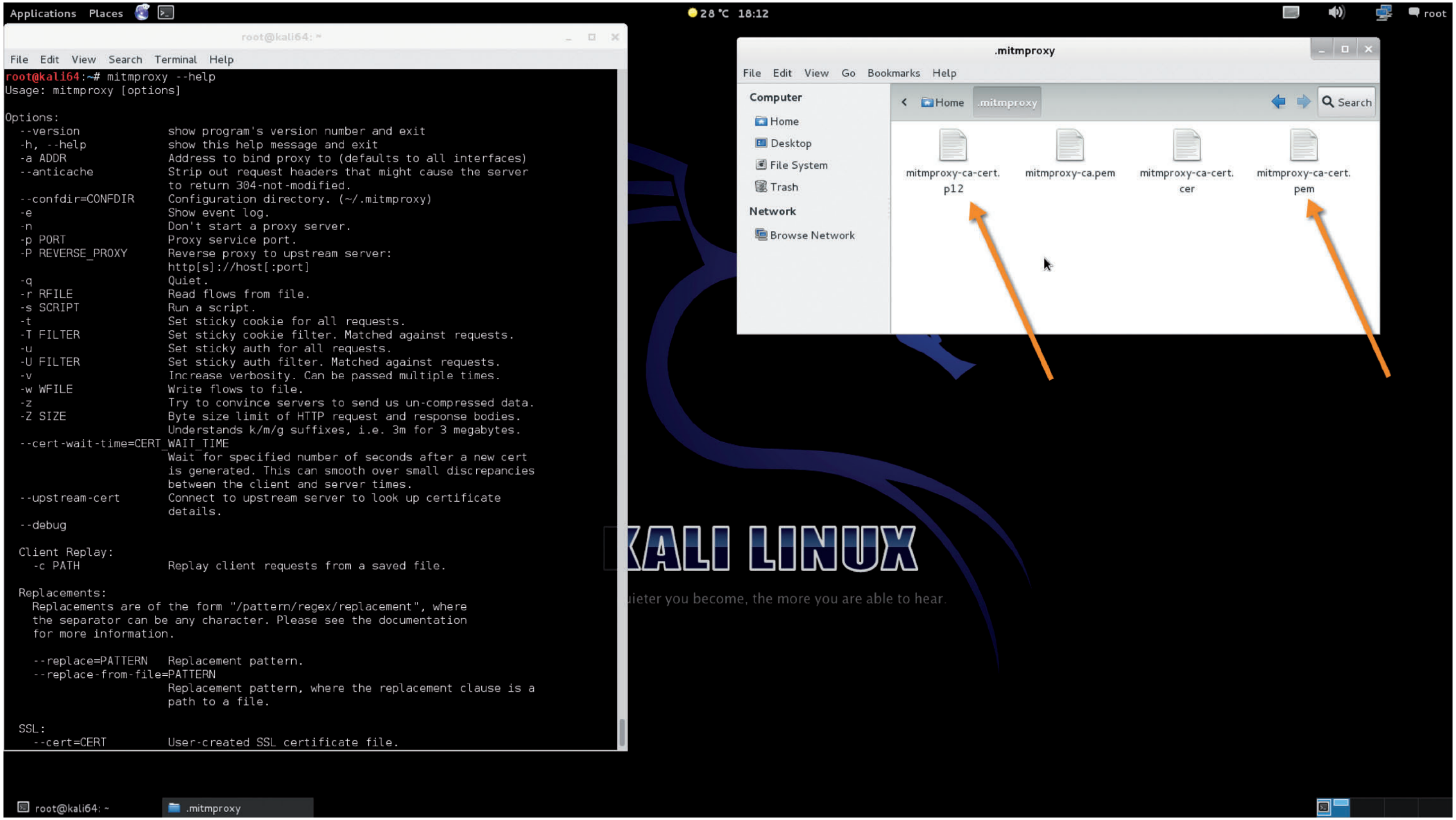
Συνεχίστε το διάβασμα.

Περί Πιστοποιητικών και Αρχών Πιστοποίησης

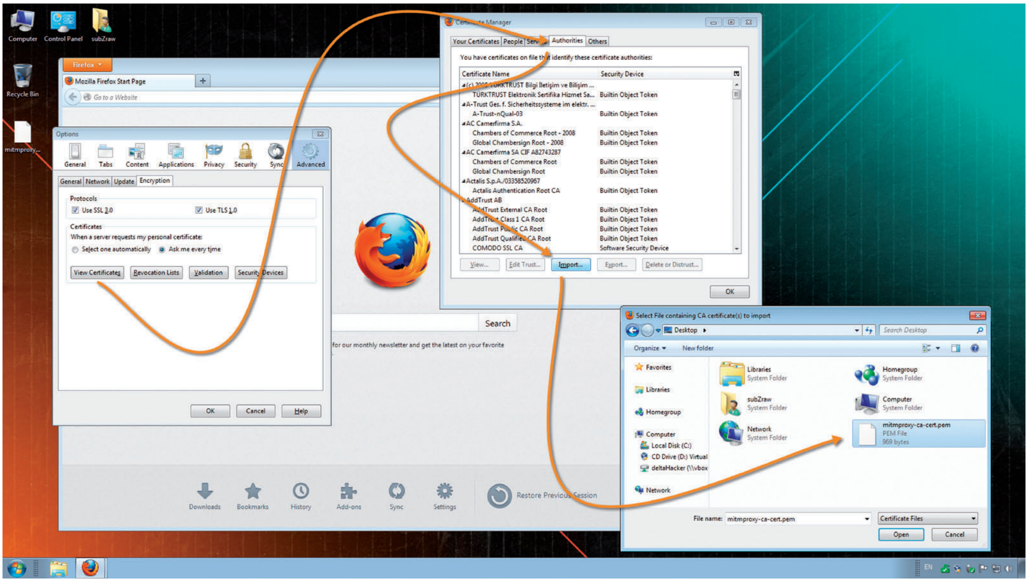
Όποτε επιχειρούμε σύνδεση σε κάποιον δικτυακό τόπο που υποστηρίζει το HTTPS, όπως, π.χ., συμβαίνει με το site μιας οποιασδήποτε τράπεζας, ξεκινά μια επικοινωνία μεταξύ των εμπλεκομένων μερών, δηλαδή του web browser και του αντίστοιχου web server, όπου με χρήση τεχνικών κρυπτογραφίας δημοσίου κλειδιού (public key encryption) τα δύο μέρη αποφασίζουν πάνω σ' ένα προσωρινό κλειδί (key). Η μετέπειτα μεταξύ τους επικοινωνία είναι κρυπτογραφημένη και το προσωρινό κλειδί είναι ό,τι χρειάζεται για την κρυπτογράφηση ή αποκρυπτογράφηση των δεδομένων σε κάθε άκρο (symmetric key encryption).

Υπάρχει όμως ένα πρόβλημα εδώ: Στα πρώτα στάδια της επικοινωνίας μεταξύ web browser και web server, πώς ο πρώτος διασφαλίζει την ταυτότητα του δεύτερου; Στη θέση του site της τράπεζας, για παράδειγμα, θα μπορούσε να βρίσκεται ένα κακόβουλο site που της μοιάζει, αλλά αντί να παρέχει υπηρεσίες web banking απλά ψαρεύει στοιχεία σύνδεσης σε λογαριασμούς πελατών της αληθινής τράπεζας.

Το συγκεκριμένο πρόβλημα ταυτοποίησης λύνεται με τη χρήση των λεγό-



Την πρώτη φορά που τρέχουμε το mitmproxy, το πρόγραμμα δημιουργεί στον κατάλογο ~/.mitmproxy ορισμένα αρχεία που του χρησιμεύουν προκειμένου να λειτουργεί ως CA. Μεταξύ αυτών είναι τα mitmproxy-ca-cert.pem, για εγκατάσταση στους web browsers, και mitmproxy-ca-cert.p12, για εγκατάσταση στα ίδια τα Windows. Κατά τις δοκιμές μας εγκαταστήσαμε το mitmproxy-ca-cert.pem στον Firefox.



Για την προσθήκη του mitmproxy CA στη λίστα των γνωστών CAs που εμπιστεύεται ο Firefox, ακολουθούμε τη διαδρομή Options → Advanced → Encryption → View Certificates → Authorities → Import. Στο παράθυρο επιλογής αρχείων που εμφανίζεται υποδεικνύουμε το αρχείο mitmproxy-ca-cert.pem, το οποίο έχουμε μεταφέρει στο σύστημα από τον κατάλογο ~/.mitmproxy, στο box που κατοικεί το mitmproxy. Στο επόμενο παράθυρο που θα εμφανιστεί τσεκάρουμε την επιλογή Trust this CA to identify websites και κάνουμε ένα κλικ στο κουμπί OK. Αυτό ήταν, πλέον ο Firefox γνωρίζει άλλη μία έμπιστη Αρχή Πιστοποίησης!

μενων ψηφιακών πιστοποιητικών (digital certificates). Το web site κάθε τράπεζας ή οποιουδήποτε άλλου δικτυακού τόπου που μιλάει HTTPS, λοιπόν, διαθέτει κι από ένα πιστοποιητικό. Αυτό περιλαμβάνει τα στοιχεία του ιδιοκτήτη, του οργανισμού ή της εταιρείας στην οποία ανήκει το site, καθώς και το λεγόμενο δημόσιο κλειδί (public key) του site. Κάθε web browser μπορεί κι εξετάζει τα περιεχόμενα των ψηφιακών πιστοποιητικών, αλλά αυτό από μόνο του δεν είναι αρκετό. Τα ψηφιακά πιστοποιητικά, βλέπετε, εκδίδονται εύκολα από κάθε ενδιαφερόμενο. Έτσι, μπορώ εγώ να εκδώσω για 'μένα ένα πιστοποιητικό που να λέει ότι είμαι η Διαπλανητική Τράπεζα, το Monkey Island Bureau of War ή το υπουργείο τουρισμού του Ανατολικού Τιμόρ. Πώς λοιπόν ο επισκέπτης του site μου, που διαβάζει το πιστοποιητικό μου, είναι σίγουρος για την εγκυρότητά του;

Κάπου εδώ μπαίνουν στο παιχνίδι οι λεγόμενες Αρχές Πιστοποίησης (Certification Authorities, CAs), όπως είναι οι VeriSign, Trustwave, Comodo, GlobalSign, Rapid SSL κ.ά. Κάθε CA αποτελεί μια έμπιστη οντότητα και δουλειά της είναι να επικυρώνει την εγκυρότητα των ψηφιακών πιστοποιητικών, υπογράφοντάς τα. Κάθε σύγχρονος web browser διαθέτει μια λίστα με όλες τις γνωστές Αρχές Πιστοποίησης. Έτσι, με το που βλέπει ένα ψηφιακό πιστοποιητικό που είναι υπογεγραμμένο από κάποια εξ αυτών, καταλαβαίνει ότι είναι έγκυρο και συνεπώς εμπιστεύεται τη γνησιότητα του αντίστοιχου site.

Παρά τα προβλήματα που κατά καιρούς έχουν προκύψει στο οικοσύστημα των CAs, η όλη ιδέα γενικά δουλεύει.

Βεβαίως, όπως πάντα, στην ιστορία υπάρχει κι ένα «εκτός αν».

Proxies σαν όλους τους άλλους #NOT

Το «εκτός αν» της δικής μας ιστορίας έγκειται σε μια επίθεση MiTM, κατά την οποία μεταξύ web browser και web server βρίσκεται ένας proxy. Αυτός ο proxy δεν ενδιαφέρεται για το φιλτράρισμα ή την επιτάχυνση της πρόσβασης των clients στο web. Πέρα απ' όλα αυτά, βασική του απο-

στολή είναι να παράγει ψηφιακά πιστοποιητικά κατά τρόπο δυναμικό, με απώτερο σκοπό να αποκρυπτογραφεί *εκείνος* τις συνδέσεις HTTPS — εννοείται πριν τα δεδομένα να φτάνουν στους αντίστοιχους web servers.

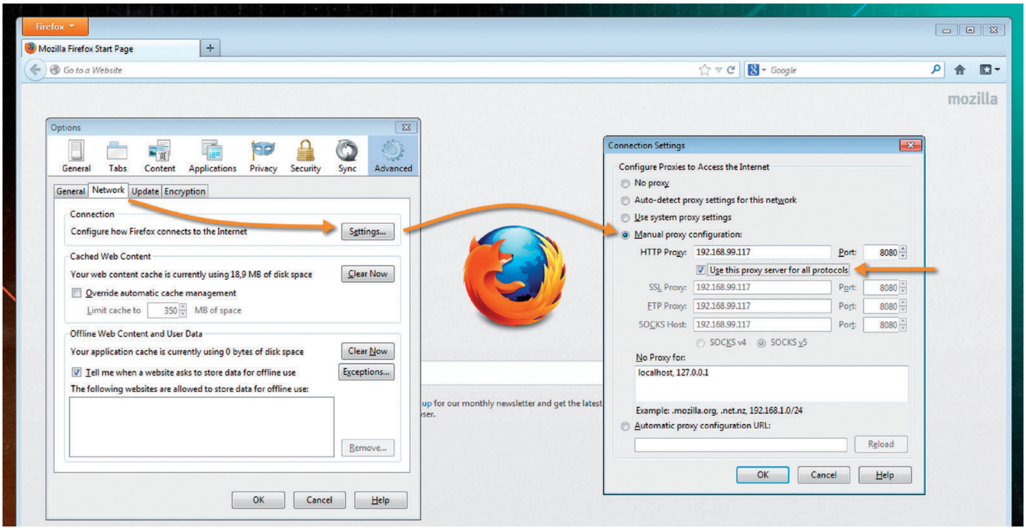
Ο τρόπος που πετυχαίνει το σκοπό του είναι απλός κι αποτελεσματικός. Ας υποθέσουμε ότι η δικτυακή κίνηση του υπολογιστή μου —ή έστω μόνο του αγαπημένου μου web browser— περνάει μέσα απ' αυτόν τον proxy (το γιατί και το πώς θα τα συζητήσουμε στη συνέχεια). Πάω λοιπόν κάποια στιγμή να επισκεφτώ το Gmail, μέσω HTTPS φυσικά. Ο proxy βλέπει το αίτημά μου κι αμέσως παράγει ένα ψηφιακό πιστοποιητικό, το οποίο φαίνεται να προέρχεται από τον δικτυακό τόπο που θέλω να επισκεφτώ, δηλαδή από το Gmail. Η δουλειά που έχει να κάνει ο proxy είναι αρκετά πιο περίπλοκη απ' όσο αρχικά δείχνει, ωστόσο δεν είναι κάτι το αδύνατο. Το ψευδεπίγραφο πιστοποιητικό το υπογράφει ο ίδιος ο proxy κι ακολούθως το παρουσιάζει στον web browser. Εκείνος το δέχεται — κι ας μην είναι υπογεγραμμένο από κάποιον έμπιστο CA! Το πώς είναι δυνατόν να συμβεί κάτι τέτοιο θα το δούμε σε πολύ λίγο. Από τη στιγμή λοιπόν που ο web browser έχει αποδεχθεί το πιστοποιητικό του proxy, ο τελευταίος είναι σε θέση να αποκρυπτογραφεί τη σύνδεση προς το Gmail και να διαβάζει τα δεδομένα που αποστέλλονται προς τον αντίστοιχο web server. Ταυτόχρονα, μιλάει στον τελευταίο παριστάνοντας τον πελάτη. Τα δεδομένα λοιπόν που στέλνω στο Gmail, ο proxy τα αποκρυπτογραφεί, τα καταγράφει, τα κρυπτογραφεί ξανά και τα κατευθύνει στον κανονικό τους προορισμό. Στην αντίθετη κατεύθυνση, ο proxy λαμβάνει τις απαντήσεις και τα δεδομένα που στέλνει το Gmail —μην ξεχνάτε ότι για τον web server ο proxy είναι ένας κανονικότατος πελάτης—, τα αποκρυπτογραφεί, τα καταγράφει, τα κρυπτογραφεί ξανά και τελικά τα προωθεί στον δικό μου web browser!

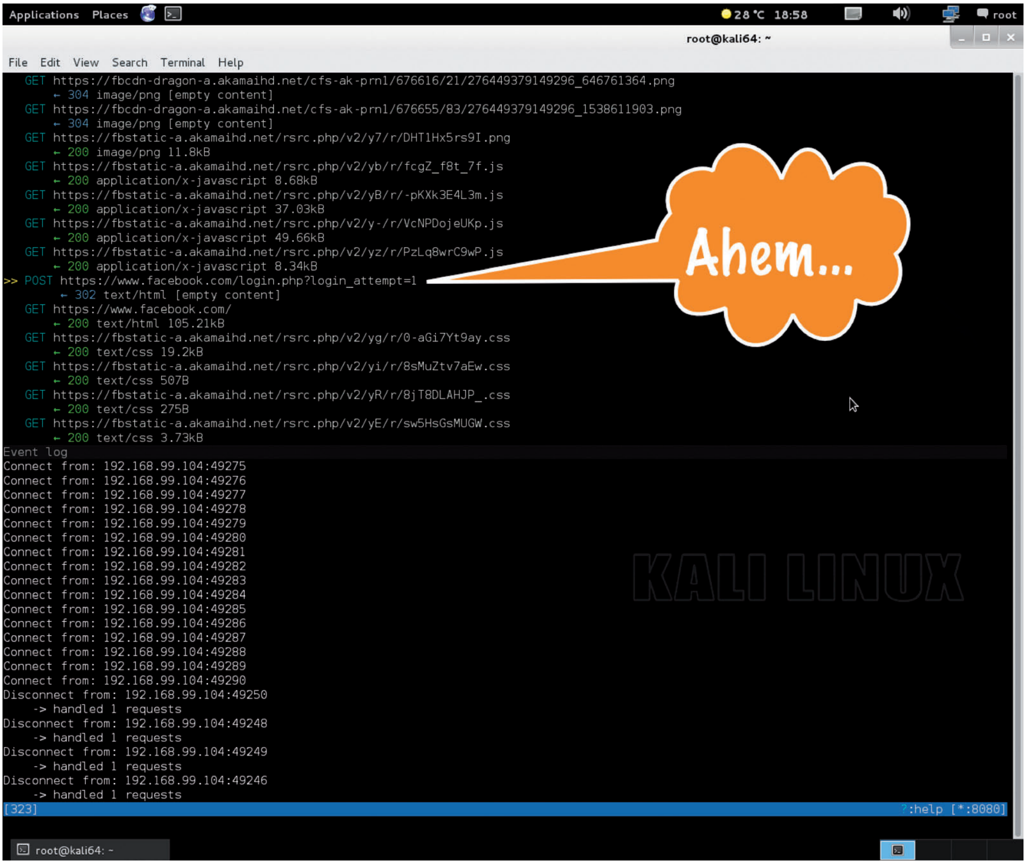
Είναι ώρα να δούμε στην πράξη πώς δουλεύει μια τέτοια επίθεση MiTM.

Τρομακτικο-διασκεδαστική δοκιμή

Το εργαλείο που χρησιμοποιούμε ονομάζεται mitmproxy και είναι σχεδιασμένο για τη δουλειά που περιγράψαμε μόλις (`http://mitmproxy.org`). Είναι γραμμένο στη γλώσσα Python και λειτουργεί σε οποιοδήποτε λειτουρ-

Για τις δοκιμές μας ρυθμίσαμε τον Firefox ώστε να χρησιμοποιεί ως proxy το VM στο οποίο έτρεχε το mitmproxy (Options → Network → Settings → Manual proxy configuration). Σημειώστε ότι το mitmproxy εξ ορισμού χρησιμοποιεί το port 8080. Παρατηρήστε επίσης ότι τσεκάραμε και την επιλογή Use this proxy server for all protocols. Τώρα, αντί να ορίζουμε «χειροκίνητα» τον proxy στους web browsers, την κίνηση HTTPS των οποίων θέλουμε να παρακολουθούμε, μπορούμε να στήσουμε έναν router για όλο το τοπικό μας δίκτυο. Ο router θα εφαρμόζει transparent proxying προς το mitmproxy, για όλα τα μηχανήματα του τοπικού δικτύου. Περισσότερα επ' αυτού διαβάστε στο `http://deltahacker.gr/?p=8868`.





Το mitmproxy λειτουργεί και, στο πάνω του μέρος, όπου παρουσιάζεται η ροή των requests από και προς τον εκάστοτε server, βλέπουμε ένα ενδιαφέρον POST προς το Facebook. Με τα βελάκια του πληκτρολογίου μετακινούμαστε στη γραμμή του POST και πατάμε το [Enter].

γικό σύστημα υπάρχει ο interpreter της γλώσσας. Υποστήριξη, ωστόσο, παρέχεται μόνο για Linux και BSD. Εμείς το δοκιμάσαμε στο Kali Linux, μιας κι εκεί είναι ήδη εγκατεστημένο. Εργαστήκαμε σε εικονικό περιβάλλον και συγκεκριμένα με δύο VirtualBox VMs: Το ένα με Windows 7 και το άλλο με το Kali. Όταν το mitmproxy εκτελείται, εξ ορισμού λειτουργεί ως ένας τυπικός proxy που ακούει για αιτήσεις πελατών στο port 8080 του localhost. Για να χρησιμοποιηθεί ο proxy από τον web browser ενός χρήστη σε ρόλο θύματος, πρέπει είτε να δηλωθεί «χειροκίνητα» στις ρυθμίσεις του browser είτε ο υπολογιστής με το mitmproxy **α)** να λειτουργεί ως router για εκείνον του θύματος και φυσικά **β)** να πραγματοποιεί το λεγόμενο transparent proxying (π.χ., με τους κατάλληλους κανόνες στο iptables ή στο pf).

Για τις δοκιμές μας, στο VM του θύματος δηλώσαμε τον proxy χειροκίνητα. Αν τώρα εσείς ενδιαφέρεστε για το στήσιμο router με ενεργοποιημένο το transparent proxying για το mitmproxy, δείτε το σχετικό μας άρθρο στο site του περιοδικού:

`http://deltahacker.gr/?p=8868`

Το mitmproxy δημιουργεί πιστοποιητικά κατ' απαίτηση, τα οποία και υπογράφει. Λειτουργεί, δηλαδή, ως Αρχή Πιστοποίησης. Επειδή όμως στον πλανήτη Γη δεν υπάρχει λειτουργικό σύστημα ή web browser που να



Το mitmproxy σε όλο του το μεγαλείο! Από τη στιγμή που μπορεί να αποκρυπτογραφεί δυναμικά το HTTPS traffic, αναλύει το POST request και μεταξύ άλλων αποκαλύπτει το username (email) και password που έχει το θύμα στο Facebook! (Το συγκεκριμένο password το χρησιμοποιούσε ο γράφων — τουλάχιστον μέχρι τη στιγμή που πήρε το screenshot :D)

εμπιστεύεται κάποιον CA ονόματι «mitmproxy», για τις δοκιμές μας οφείλουμε να ενημερώσουμε κατάλληλα τη λίστα με τους CAs του λειτουργικού συστήματος ή/και του εκάστοτε web browser. Προτείνουμε να μην κάνετε κάτι παρόμοιο στο λειτουργικό ή στον web browser ενός φυσικού υπολογιστή. Όπως εμείς, εργαστείτε σε εικονικό περιβάλλον.

Λογικό είναι εδώ να σκεφτείτε ότι κανείς σόφρων χρήστης δεν θα προσέθετε στο λειτουργικό ή στον web browser έναν CA σαν αυτόν του mitmproxy. Δεν έχετε κι άδικο, μόνο που παραβλέπετε τους υπολογιστές που βρίσκονται σε εταιρείες, εργαστήρια, οργανισμούς κ.ο.κ. Συνήθως, στους υπολογιστές αυτούς οι χρήστες δεν έχουν administrative rights και δεν μπορούν να κάνουν πολλά στο σύστημα ή/και στο λογισμικό. Αν λοιπόν κάποιος τρίτος έχει εκ των προτέρων εγκαταστήσει έναν «παράξενο» CA, τότε ακόμη κι αν τον ψάξουν και τον εντοπίσουν, οι χρήστες μας δεν θα είναι σε θέση να τον αφαιρέσουν!

Στις δοκιμές που κάναμε με το mitmproxy είδαμε πόσο τρομακτικά εύκολο είναι να υποκλαπούν προσωπικές πληροφορίες κι ευαίσθητα δεδομένα, όταν στο τοπικό δίκτυο υπάρχει κάποιο εργαλείο σαν το mitmproxy και οι χρήστες δεν έχουν απόλυτο έλεγχο στους υπολογιστές τους. Δείτε στα σχετικά screenshots μερικά ενδιαφέροντα στιγμιότυπα των δοκιμών μας.

Μέτρα προστασίας

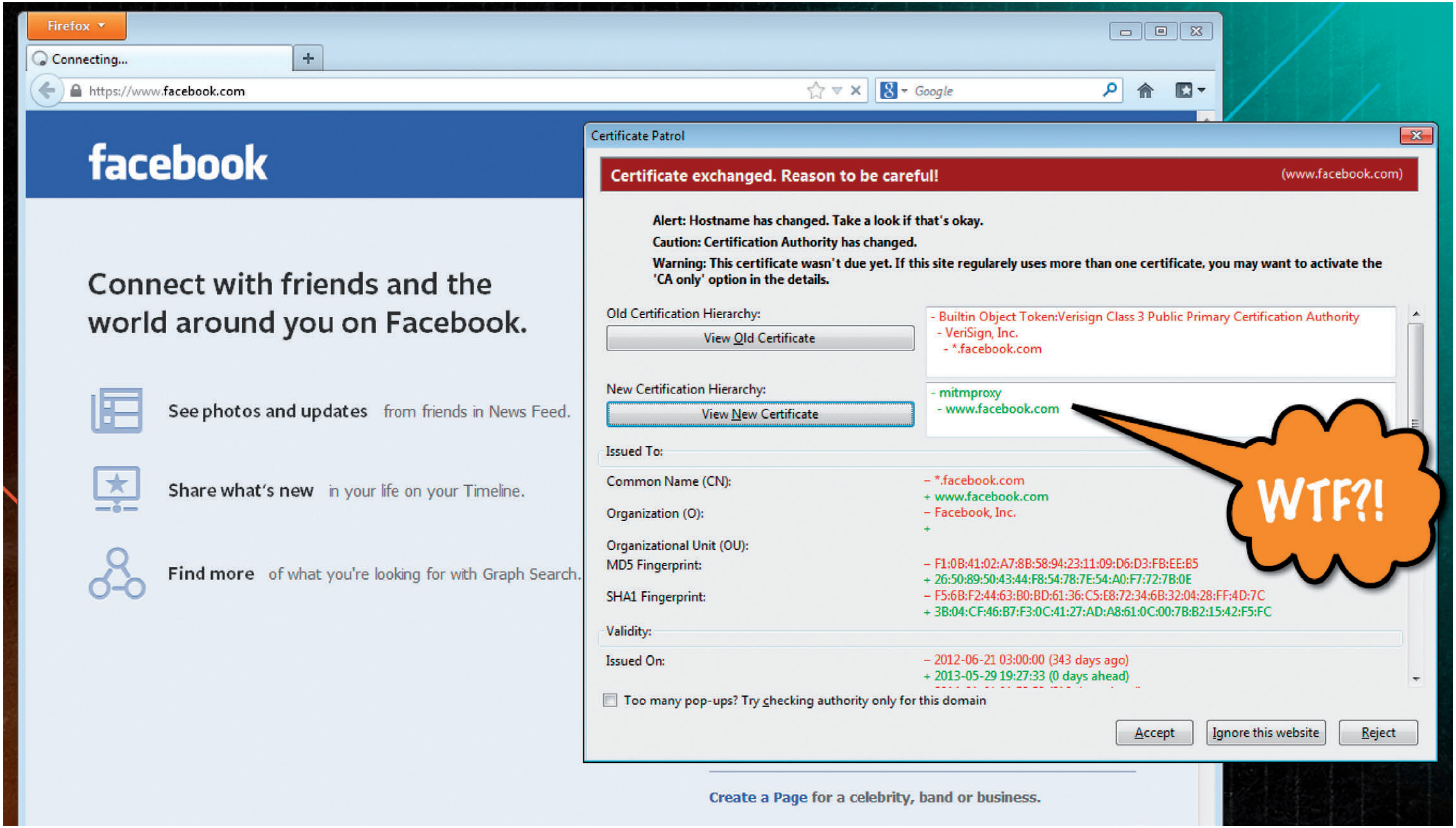
Πιστεύουμε ότι αφού παίξετε κι εσείς με το mitmproxy, στο εξής θα είστε προσεκτικοί ακόμη κι εντός του δικού σας τοπικού δικτύου. Αν μη τι άλλο, πλέον θα προσέχετε περισσότερο ποιον αφήνετε να κάθεται μπροστά από τον υπολογιστή σας ή σε ποιον δίνετε το laptop σας.

Αν τώρα μιλάμε για ένα δίκτυο του οποίου τον έλεγχο δεν έχετε και, πολύ περισσότερο, για εργασία μέσα από περιορισμένο user account, η αλήθεια είναι ότι δεν μπορείτε να κάνετε πολλά — παρά μόνο ίσως την εργασία που πληρώνεστε για να κάνετε :) Καλού κακού, πάντως, αν έχετε τη δυνατότητα εγκαταστήστε στον αγαπημένο σας web browser ένα add-on σαν το Certificate Patrol (`http://patrol.psyced.org`). Κάθε φορά που επισκέπτεστε ένα ασφαλές site, το εν λόγω add-on αποθηκεύει το αντίστοιχο πιστοποιητικό. Αν σε επόμενη επίσκεψη στο ίδιο site διαπιστωθεί ότι παρέχεται διαφορετικό πιστοποιητικό, το Certificate Patrol θα σταματήσει τη σύνδεση και θα σας ειδοποιήσει για την αλλαγή, εμφανίζοντας σχετικό παράθυρο. Έτσι, θα είστε σε θέση να γνωρίζετε ότι κάτι παράξενο συμβαίνει στο δίκτυο μέσα από το οποίο βγαίνετε στο Internet.

Τέλος, προτείνουμε ν' αφιερώσετε το χρόνο και να στήσετε κάπου τον δικό σας OpenVPN server. Κάθε φορά που χρησιμοποιείτε υπολογιστή ή/και βρίσκεστε σε δίκτυο που *δεν* εμπιστεύεστε, αν έχετε τη δυνατότητα βγαίνετε στον έξω κόσμο μέσω του προαναφερθέντος OpenVPN server. Περισσότερα επ' αυτού στο σχετικό άρθρο του deltaHacker 007 (βλ. σελ. 12) ή στο site του περιοδικού:

`http://deltahacker.gr/?p=5669`

Αυτά προς το παρόν :D



Κάθε φορά που επισκέπτεστε ένα ασφαλές site, το Certificate Patrol για τον Firefox αποθηκεύει το αντίστοιχο πιστοποιητικό. Αν σε επόμενη επίσκεψη στο ίδιο site διαπιστωθεί ότι παρέχεται διαφορετικό πιστοποιητικό, το Certificate Patrol θα σταματήσει τη σύνδεση και θα ειδοποιήσει για την αλλαγή, εμφανίζοντας σχετικό παράθυρο!



Skill: Beginner

Tags: Wireshark, tshark, streams, filters, graphs, ARP poisoning

[μέρος 2]

ΑΝΑΛΥΣΗ ΠΑΚΕΤΩΝ

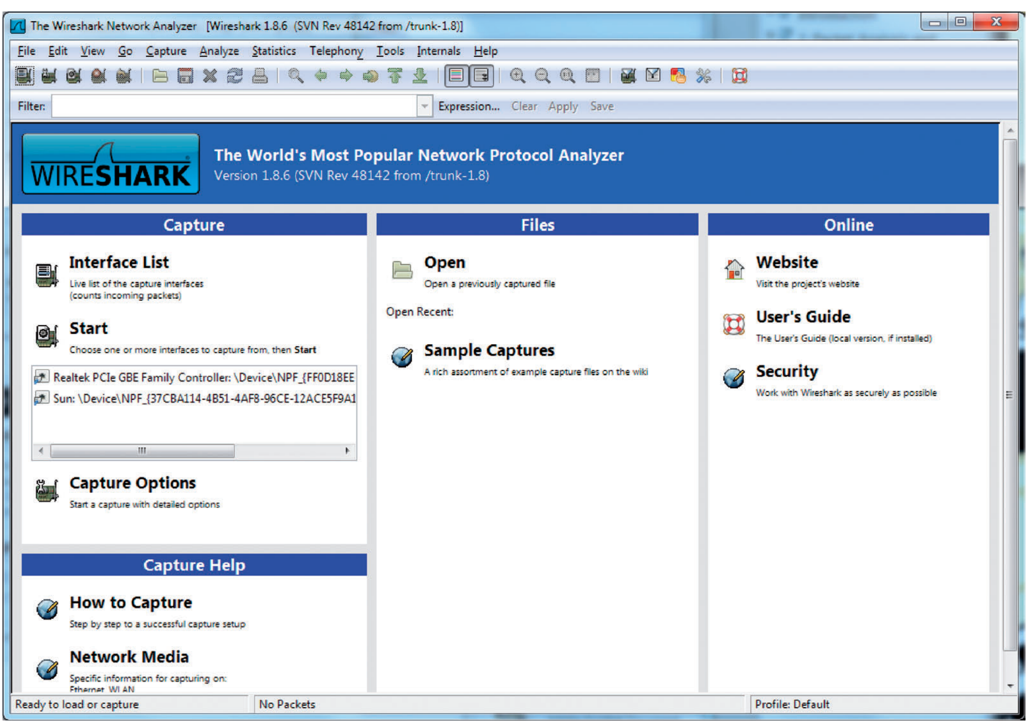
WIRESHARK

by Ion

Στο πρώτο μέρος της μίνι σειράς μας, στο deltaHacker 019, ασχοληθήκαμε με το tcpdump. Πρόκειται για ένα εργαλείο με 25 χρόνια ιστορίας, το οποίο κάλυψε πολλές από τις ανάγκες μας σχετικά με την ανάλυση των δικτυακών πακέτων. Στο παρόν άρθρο επεκτείνουμε το οπλοστάσιό μας γνωρίζοντας ακόμα ένα εξαιρετικά χρήσιμο εργαλείο: το Wireshark!

Όπως και το tcpdump, το Wireshark κουβαλάει μια ενδιαφέρουσα ιστορία. Η πρώτη έκδοση του προγράμματος αναπτύχθηκε το 1998 από έναν μεταπτυχιακό φοιτητή του Πανεπιστημίου του Μιζούρι, και η αρχική του ονομασία ήταν Ethereal. Κάποιοι μεγαλύτεροι σε ηλικία ενδέχεται να το θυμόσαστε ακόμη με αυτό το όνομα. Οκτώ χρόνια μετά την πρώτη έκδοση του προγράμματος κι ενώ ο ίδιος προγραμματιστής και η ομάδα των εθελοντών που τον πλαισίωναν συνέχιζαν να αναπτύσσουν το πρόγραμμα, το project άλλαξε όνομα λόγω πνευματικών δικαιωμάτων. Έτσι, στα μέσα του 2006 γεννήθηκε το Wireshark. Από τότε συνεχίζει να αναπτύσσεται ενεργά και οι προγραμματιστές που έχουν συνεισφέρει κώδικα για τη λειτουργία έχουν ξεπεράσει τους 600! Η επίσημη διεύθυνση του προγράμματος βρίσκεται στο `www.wireshark.org` και από εκεί είναι διαθέσιμη και η πλέον πρόσφατη έκδοση του.

Το Wireshark εκδίδεται υπό την άδεια χρήσης GNU GPL, που με απλά λόγια σημαίνει ότι ο κώδικάς του είναι δωρεάν κι ανοικτός στον καθένα. Πέρα από το (ανύπαρκτο) κόστος, το Wireshark προσφέρει και κάποια άλλα πλεονεκτήματα που το καθιστούν ένα από τα καλύτερα εργαλεία στο είδος του. Για παράδειγμα, υποστηρίζει ένα μεγάλος πλήθος πρωτοκόλλων, που ξεπερνούν τα 850. Σε αυτά συμπεριλαμβάνονται όλα τα γνωστά κι ευρέως χρησιμοποιούμενα δικτυακά πρωτόκολλα, όπως το IP, το ARP, το DHCP κ.ά., καθώς κι εμπορικά πρωτόκολλα όπως το AppleTalk ή το BitTorrent. Ενδιαφέρουσες πληροφορίες σχετικά με καθένα από τα υποστηριζόμενα πρωτόκολλα μπορείτε να βρείτε στη σελίδα `http://wiki.wireshark.org/ProtocolReference`. Το Wireshark υποστηρίζει όλα τα γνωστά λειτουργικά συστήματα (Linux/Unix, Windows και OS X) και είναι αρκετά φιλικό προς το χρήστη. Εκτός από το interface, που είναι αρκετά εύκολο στη χρήση, προσφέρει και διάφορα έξτρα καλούδια, όπως



Το λιτό περιβάλλον του Wireshark είναι αρκετό για να ξεκινήσουμε. Η στήλη στα αριστερά αποτελεί το σημείο εκκίνησης για ένα packet capture.

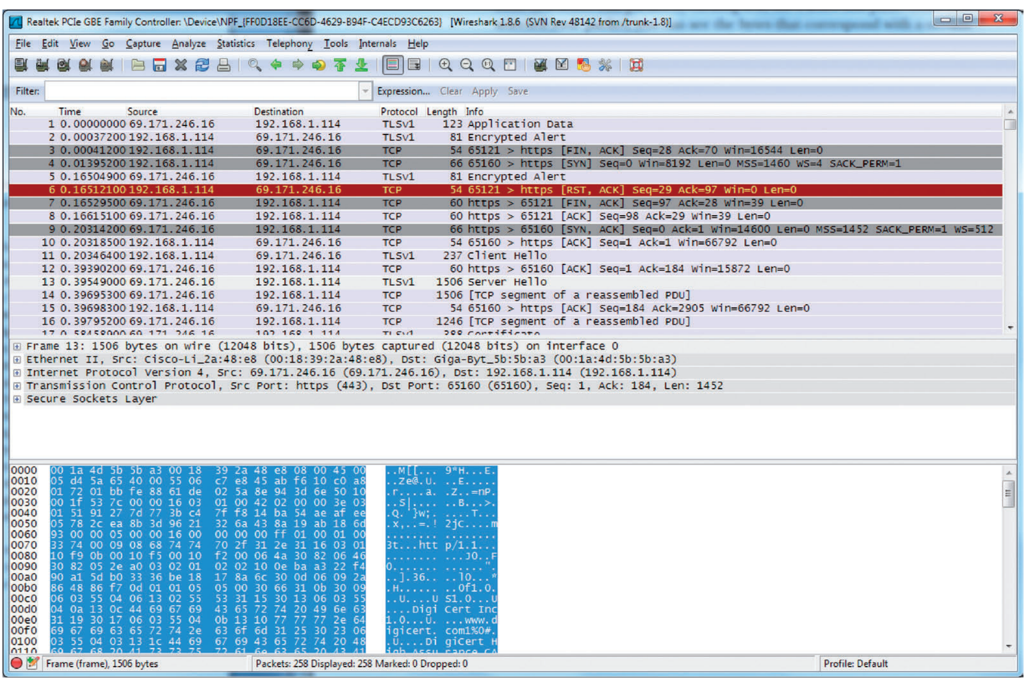
το χρωματισμό των πακέτων ανάλογα με το πρωτόκολλο, καθώς και τη γραφική αναπαράσταση των δεδομένων.

Πέρα από το ίδιο το Wireshark αξίζει να γίνει μια αναφορά και στο Tshark, που είναι στην ουσία η έκδοση του Wireshark για τη γραμμή εντολών κι αποτελεί, θα λέγαμε, το ξαδερφάκι του. Το Tshark συμπεριφέρεται σαν το tcpdump και μπορείτε να το χρησιμοποιήσετε ως ένα πιο σκληροπυρηνικό εργαλείο για ανάλυση πακέτων. Είναι γνωστή εξάλλου η συμπάθεια μας προς το ρητό «real men do it command-line» :D Αρκετά όμως με την εισαγωγή. Νομίζουμε πως ήρθε η ώρα να περάσουμε στην πράξη.

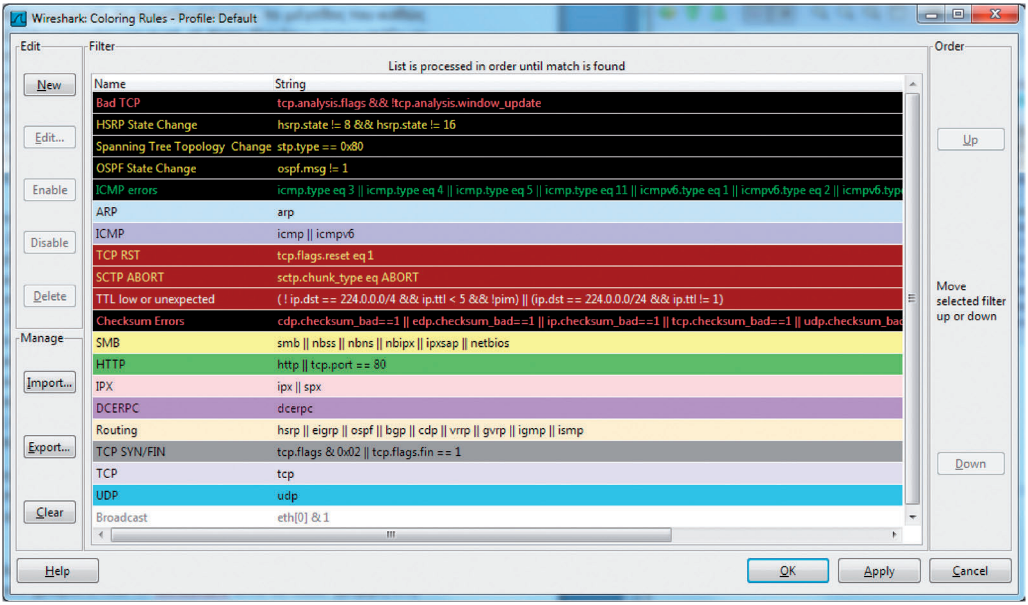
Τα βασικά

Στο παρόν άρθρο και για την παραθυρική έκδοση του Wireshark θα δουλέψουμε λίγο στα Windows. Αφενός δεν θέλουμε να αφήσουμε ορισμένους φίλους μας παραπονεμένους, αφετέρου θα ήταν καλό να φανεί ότι η ανάλυση πακέτων ενός δικτύου δεν είναι κάτι που μπορούν να εκτελούν μόνο οι πιο «ψαγμένοι» χρήστες[1]. Επομένως, ξεκινάμε κατεβάζοντας την τελευταία έκδοση του Wireshark από τη διεύθυνση `www.wireshark.org/download.html`. Κατά τη συγγραφή του άρθρου, η πιο πρόσφατη έκδοση ήταν η 1.8.6. Για τις ανάγκες μας θα δουλέψουμε σε έναν φυσικό υπολογιστή με Windows 7 32bit. Η διαδικασία εγκατάστασης του Wireshark είναι απλή και ακολουθεί τη γνωστή φιλοσοφία του Next → Next → Next. Κατά τη διάρκειά της θα σάς ζητηθεί να εγκαταστήσετε και τη βιβλιοθήκη WinPcap. Το πρόγραμμα εγκατάστασης για την εν λόγω βιβλιοθήκη είναι ενσωματωμένο στο αρχείο εγκατάστασης του Wireshark και δεν χρειάζεται καμία πρόσθετη ενέργεια από μέρους σας. Όπως καταλαβαίνετε, πρόκειται για την εκδοχή της βιβλιοθήκης libpcap, που συναντάμε στο Unix, για Windows.

Το βασικό παράθυρο του Wireshark, όπου εμφανίζονται όλα τα καταγεγραμμένα πακέτα, θα σας συντροφεύσει για πολλές από τις ώρες που έρχονται (το ευχόμαστε).



1. Αν και πλέον η εγκατάσταση μιας διανομής Linux έχει γίνει παιχνιδάκι!



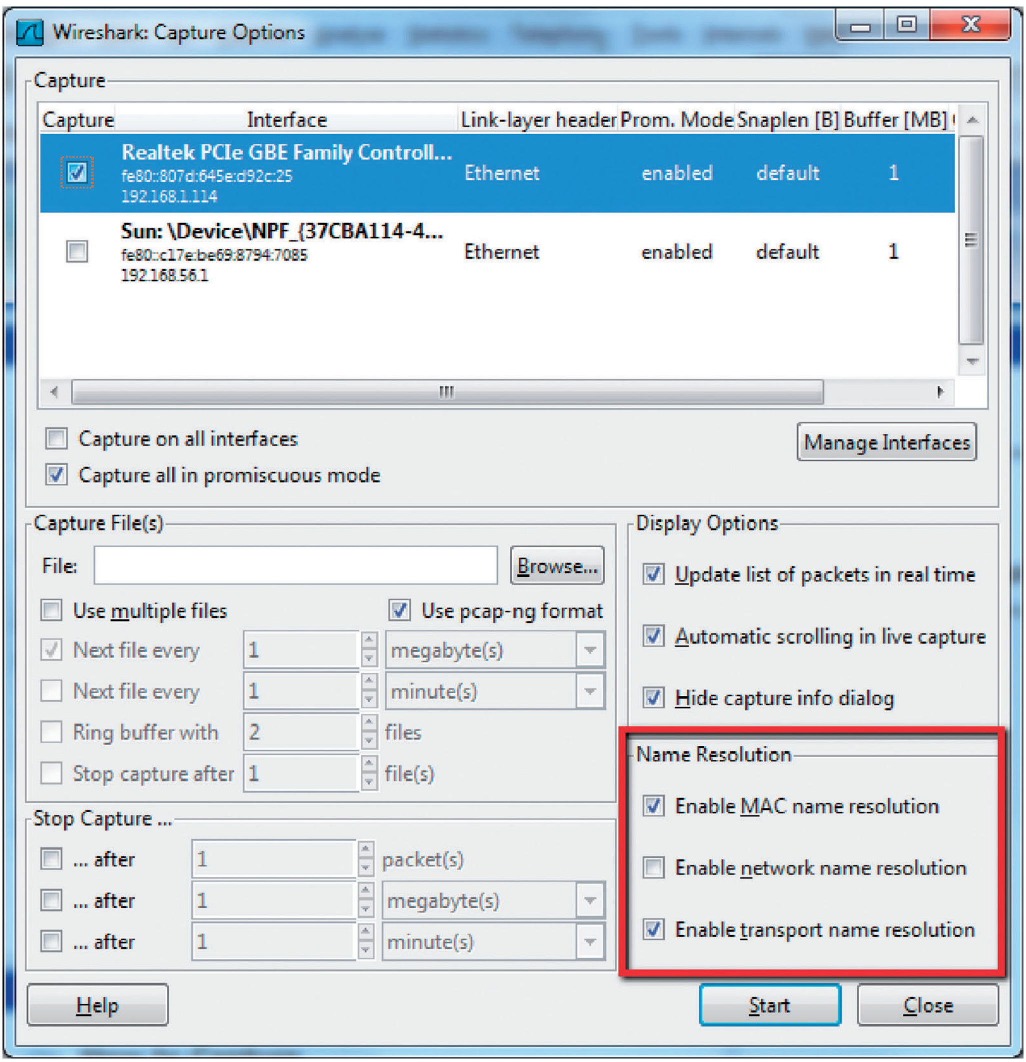
Βάζουμε λίγο χρώμα στη ζωή μας, με τα πολύ ωραία coloring rules του προγράμματος. Θα μας βοηθήσουν να αποκτήσουμε μια γρήγορη εικόνα για τη φύση των πακέτων που καταγράφουμε.

Ανοίγοντας για πρώτη φορά το Wireshark παρουσιάζεται ένα σχετικά λιτό παράθυρο, το οποίο παρέχει μερικές αρχικές επιλογές. Συγκεκριμένα, βλέπουμε τρεις στήλες και για να ξεκινήσουμε την εργασία μας θα στραφούμε στην αριστερή. Εκεί, το Wireshark εμφανίζει μια λίστα με όλα τα διαθέσιμα interfaces που μπορεί να χρησιμοποιήσει. Όπως είχαμε εξηγήσει στο προηγούμενο άρθρο της σειράς, τα interfaces αποτελούν στην ουσία τις κάρτες δικτύου που διαθέτει ο υπολογιστής μας. Εδώ, επομένως, διαλέγουμε από ποια κάρτα/σύνδεση θα ακούει το Wireshark για πακέτα. Όπως μπορείτε να δείτε στη σχετική εικόνα, το δικό μας σύστημα διαθέτει δύο interfaces, από τα οποία μόνο το ένα είναι «φυσικό» και ξεχωρίζει από το όνομα της γνωστής κατασκευάστριας εταιρείας (Realtek). Εσείς, θα πρέπει να πατήσετε στη δική σας κάρτα δικτύου και στη συνέχεια να κάνετε κι ένα κλικ στο κουμπάκι Start. Συγχαρητήρια! Μόλις αρχίσατε το πρώτο σας packet capture με το Wireshark! Αν είστε συνδεδεμένοι σε κάποιο τοπικό δίκτυο, θα δείτε σύντομα να παρουσιάζονται αρκετές νέες γραμμές, στο παράθυρο που εμφανίστηκε πατώντας Start. Πρόκειται στην ουσία για το κεντρικό παράθυρο του Wireshark, στο οποίο θα ξοδεύετε το 99% του χρόνου σας όταν αναλύετε πακέτα. Το εν λόγω παράθυρο χωρίζεται σε τρεις βασικές περιοχές. Από πάνω προς τα κάτω, πρόκειται για τη «Λίστα Πακέτων», τις «Λεπτομέρειες Πακέτου» και τα «Bytes Πακέτου». Ας τις πάρουμε με τη σειρά.

- Το πρώτο τμήμα, η «Λίστα Πακέτων», αποτελεί έναν πίνακα στον οποίο εμφανίζονται όλα τα καταγεγραμμένα πακέτα. Κάθε μία γραμμή που βλέπετε αντιστοιχεί σε ένα μοναδικό πακέτο που καταγράφτηκε κατά τη συνεδρία και το οποίο έχει ορισμένα χαρακτηριστικά. Αυτά τα χαρακτηριστικά εμφανίζονται στην ίδια γραμμή, η οποία είναι χωρισμένη σε στήλες. Έτσι έχουμε πληροφορίες για τον αριθμό του πακέτου, την ώρα κατά την οποία καταγράφτηκε, τη διεύθυνση προέλευσης, τη διεύθυνση προορισμού, το πρωτόκολλο, το μέγεθος κ.ά. Η Λίστα Πακέτων παρουσιάζει μια επισκόπηση της κίνησης σε υψηλό, όπως λέμε, επίπεδο.

- Η μεσαία περιοχή προβάλλει τις λεπτομέρειες για το εκάστοτε πακέτο που έχουμε επιλέξει. Επομένως, για να δούμε *όλες* τις πληροφορίες για ένα πακέτο, πρέπει αρχικά να το επιλέξουμε από την πρώτη περιοχή. Όπως καταλαβαίνετε, αυτό το τμήμα κατεβαίνει πιο χαμηλά και βοηθά σε ένα μεσαίο επίπεδο ανάλυσης των πακέτων.
- Στην κάτω περιοχή του παραθύρου (Bytes Πακέτου) εμφανίζονται τα περιεχόμενα του επιλεγμένου πακέτου, τα οποία κατά πάσα πιθανότητα θα φαντάζουν στα μάτια σας σαν κινέζικα! Αυτό δεν πρέπει να σας προβληματίζει, αφού στη συγκεκριμένη περιοχή παρουσιάζονται τα περιεχόμενα του πακέτου σε ακατέργαστη μορφή — έτσι ακριβώς όπως μεταδίδονται στο δίκτυο. Η συγκεκριμένη περιοχή χωρίζεται κι αυτή σε στήλες. Στις πρώτες εμφανίζεται η δεκαεξαδική αναπαράσταση των δεδομένων του πακέτου, ενώ στην τελευ-

Το Name Resolution είναι απαραίτητο ώστε να μετατρέπουμε όλες τις περίεργες διευθύνσεις σε human-readable μορφή!



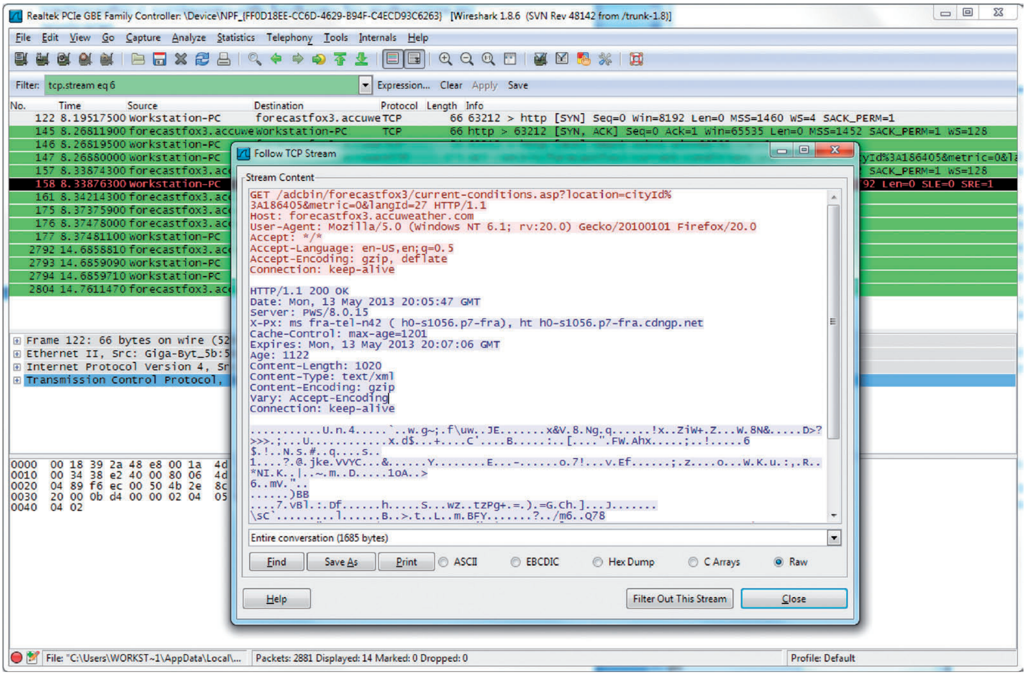
ταία στήλη βλέπουμε τα δεδομένα σε μορφή human-readable. Με λίγη προσοχή, θα διαπιστώσετε ότι σε αυτή τη στήλη εμφανίζονται διευθύνσεις και διάφορες άλλες χρήσιμες πληροφορίες.

Αν αφήσατε το Wireshark να καταγράφει όσο διαβάζατε τα παραπάνω, μπορείτε τώρα να το σταματήσετε. Αυτό γίνεται πατώντας το τέταρτο κουμπάκι, κάτω από το μενού του παραθύρου.

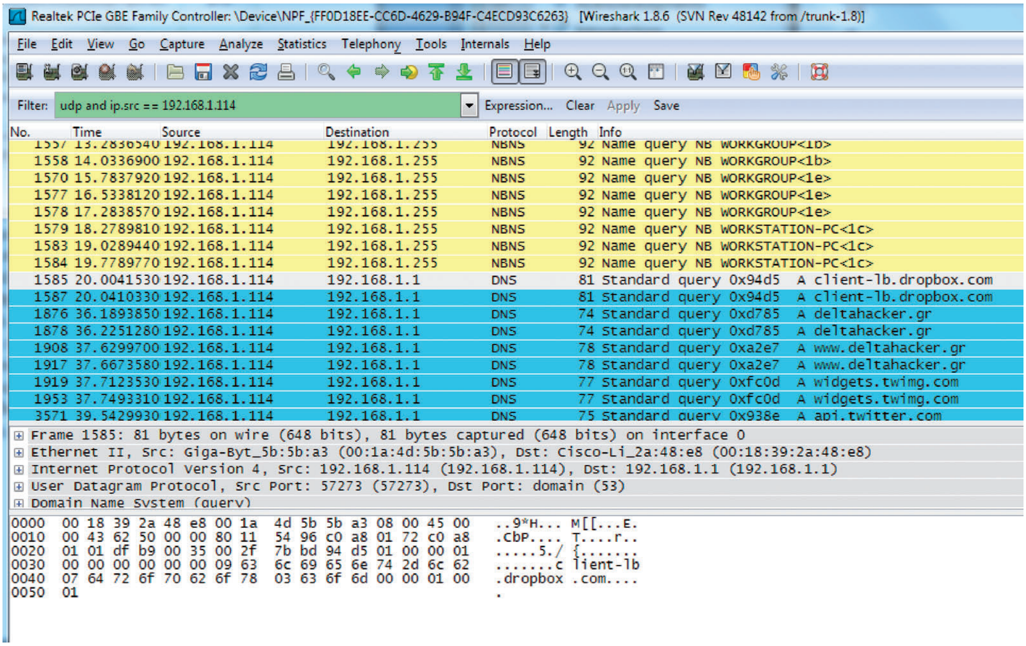
Αυτό που κάνει συνήθως εντύπωση στους νέους χρήστες είναι το πλήθος των διαφορετικών χρωμάτων που χρησιμοποιεί Wireshark στην περιοχή «Λίστα Πακέτων». Προφανώς, αυτά τα χρώματα δεν επιλέγονται τυχαία κι αποτελούν έναν εύκολο τρόπο διάκρισης των πρωτοκόλλων. Πηγαίνοντας στο View → Coloring Rules βλέπετε μια λίστα με τους κανόνες, που χρησιμοποιούνται για τη διάκριση και τον χρωματισμό των πρωτοκόλλων. Από εκεί μπορείτε να τροποποιήσετε τους κανόνες χρωματισμού ή να προσθέσετε και τους δικούς σας.

Χρήσιμες δυνατότητες

Name resolution. Μια από τις χρήσιμες δυνατότητες που διαθέτει το Wireshark και την οποία σίγουρα θα θέλετε να γνωρίζετε, είναι η επιλογή για name resolution ή «μετάφραση ονομάτων». Πρόκειται για τη μετάφραση των διευθύνσεων IP που εμφανίζονται στα καταγεγραμμένα πακέτα μιας συνεδρίας, σε μορφή human-readable. Όπως γίνεται εμφανές, τα διάφορα πρωτόκολλα χρησιμοποιούν ελαφρώς περίπλοκα αλφαριθμητικά συστήματα για την αναπαράσταση των υπολογιστών που επικοινωνούν μεταξύ τους, οπότε το name resolution μπορεί να κάνει τα πράγματα πολύ πιο απλά για τον αναλυτή. Για παράδειγμα, στις στήλες προέλευσης και προορισμού της «Λίστας Πακέτων», η διεύθυνση 109.74.192.50 θα εμφανίζεται ως deltahacker.gr.



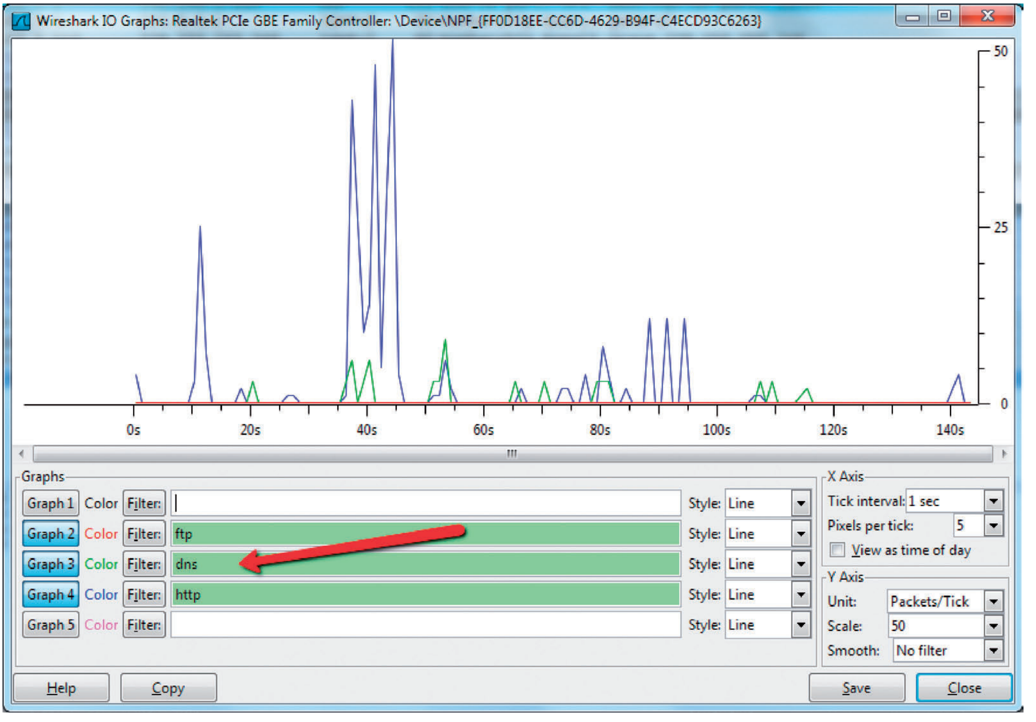
Μια από τις πολύ χρήσιμες δυνατότητες του Wireshark είναι αυτή που μας εμφανίζει τη ροή των πακέτων από και προς μία σύνδεση με έναν συγκεκριμένο server.



Για την ενεργοποίηση του name resolution ακολουθήστε τη διαδρομή Capture → Options. Στο παράθυρο που εμφανίζεται, συγκεκριμένα στο κάτω δεξιό μέρος, μπορείτε να επιλέξετε μεταξύ τριών διαφορετικών μεταφράσεων: **1)** διευθύνσεις MAC σε αντίστοιχες διευθύνσεις IP, **2)** διευθύνσεις IP σε domain names, **3)** μετάφραση των δικτυακών θυρών στο όνομα της συνήθους υπηρεσίας (π.χ., η θύρα 22 μπορεί να εμφανίζεται ως SSH και η 80 ως HTTP). Κάτι σημαντικό που πρέπει να θυμόσαστε είναι πως το name resolution πρέπει να επιλέγεται πριν ξεκινήσει η καταγραφή πακέτων, καθώς η μετάφραση κάθε διεύθυνσης δεν αποθηκεύεται αυτόματα στο αρχείο καταγραφής (pcap).

*Κάθε αναλυτής που σέβεται τον εαυτό του πρέπει να μάθει τους απλούς λογικούς κανόνες κατασκευής φίλτρων του Wireshark. Θα κάνουν την ανάλυση *πολύ* πιο εύκολη!*

Εμφάνιση ροών TCP. Πρόκειται για μία από τις πιο χρήσιμες δυνατότητες που δίνει το Wireshark. Ο καταιγισμός πληροφοριών στη «Λίστα Πακέτων» είναι πιθανό να σας προκαλέσει πονοκέφαλο, στην προσπάθεια να παρακολουθήσετε μια συγκεκριμένη αλληλουχία πακέτων μεταξύ δύο υπολογιστών. Αντί να βλέπετε όλα τα δεδομένα που ταξίδεψαν στο δίκτυο και πέρασαν από την κάρτα δικτύου σας, μπορείτε να δείτε μια λίστα με τα πακέτα που αντάλλαξαν δύο συγκεκριμένοι υπολογιστές και μάλιστα με χρονολογική σειρά. Αυτό πετυχαίνει η επιλογή Follow TCP stream, η οποία αποκτά ιδιαίτερη αξία όταν αναλύουμε πακέτα πρωτοκόλλων που στέλνουν τα δεδομένα σε μορφή plaintext (όπως, π.χ., το HTTP). Ας δούμε πώς γίνεται κάτι τέτοιο στην πράξη. **1)** Βρίσκουμε ένα πακέτο της ακολουθίας που μας ενδιαφέρει (για παράδειγμα ένα πακέτο που αφορά στο HTTP και περιέχει ένα GET request). **2)** Επιλέγοντάς το, πατάμε δεξί κλικ πάνω του και κατόπιν στην επιλογή Follow TCP Stream (ή UDP/SSL, αναλόγως). Έτσι το Wireshark θα φτιάξει αυτόματα ένα φίλτρο, με τη βοήθεια του οποίου εμφανίζονται μόνο τα πακέτα που αφορούν στη συγκεκριμένη ροή δεδομένων. Ταυτόχρονα, σε ένα νέο παράθυρο εμφανίζονται τα πακέτα που ανταλλάχθησαν, σε μορφή human-readable.



Διαλέγοντας τα είδη των πακέτων που μας ενδιαφέρουν, μπορούμε να σχηματίσουμε τα κατάλληλα γραφήματα που θα μας δώσουν μια εποπτική εικόνα για τη κίνηση του δικτύου μας.

Filters. Ίσως το πιο δυνατό χαρακτηριστικό του Wireshark και σίγουρα ένα απαραίτητο εργαλείο για να κάνουμε σωστά τη δουλειά μας! Τα φίλτρα αποτελούν λογικές εκφράσεις, με τις οποίες ορίζουμε κριτήρια διάκρισης των πακέτων. Αν τα χαρακτηριστικά ενός πακέτου πληρούν τα κριτήριά μας το πακέτο παραμένει στη «Λίστα Πακέτων». Σε αντίθετη περίπτωση κρύβεται, μέχρι να απενεργοποιήσουμε το φίλτρο. Όπως θα έχετε αντιληφθεί, ένα αρχείο pcap περιέχει χιλιάδες αν όχι εκατομμύρια πακέτα. Επομένως, η δυνατότητα δημιουργίας φίλτρων είναι εξαιρετικά χρήσιμη! Το Wireshark προσφέρει δύο ειδών φίλτρα: Αυτά που εφαρμόζονται πριν την έναρξη μιας συνεδρίας packet capturing κι αυτά που εφαρμόζονται μετά. Τα δύο είδη λειτουργούν με τον ίδιο ακριβώς τρόπο και διαφέρουν μόνο στο πώς εφαρμόζονται. Όταν ορίζουμε ένα φίλτρο πριν την έναρξη της καταγραφής, στην ουσία λέμε στο Wireshark ποια πακέτα να καταγράψει και ποια να αγνοήσει. Αυτό σημαίνει πως τα τελευταία δεν θα αποθηκευτούν καν στο αρχείο καταγραφής. Στη δεύτερη περίπτωση, τα φίλτρα που εφαρμόζονται μετά την καταγραφή καθορίζουν ποια πακέτα μας ενδιαφέρουν και θέλουμε να εμφανιστούν. Τα υπόλοιπα δεν θα εμφανιστούν καθόλου, αλλά θα παραμείνουν αποθηκευμένα στο αρχείο pcap.

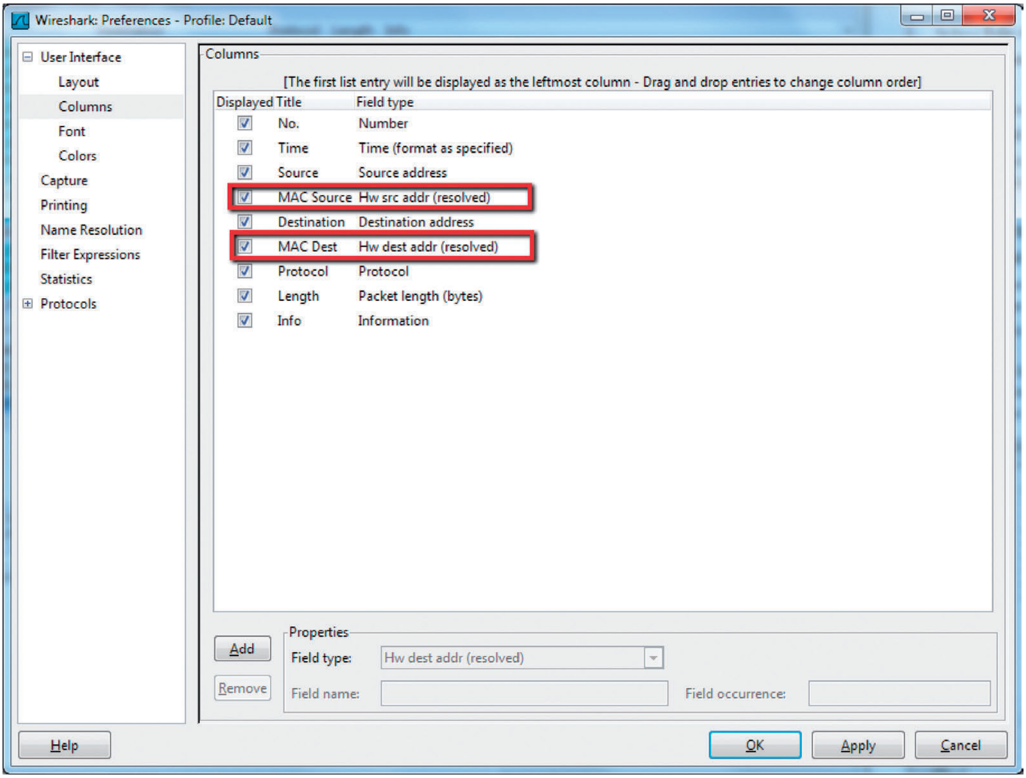
Τα φίλτρα που εφαρμόζονται κατά τη διάρκεια της καταγραφής μπορούμε να τα ενεργοποιήσουμε από τη διαδρομή Capture → Options. Εκεί, αρκεί να κάνουμε διπλό κλικ πάνω στο interface που επιθυμούμε και να συμπληρώσουμε κατάλληλα το πεδίο Capture filter, στο νέο παράθυρο που εμφανίζεται. Ακολούθως, αποθηκεύουμε την επιλογή μας και μπορούμε να ξεκινήσουμε την καταγραφή, πατώντας στο κουμπί Start. Τα φίλτρα αυτού του είδους χρησιμοποιούνται συνήθως σε περιπτώσεις που

θέλουμε να μελετήσουμε τη δικτυακή συμπεριφορά μας συγκεκριμένης υπηρεσίας. Για παράδειγμα, αν γνωρίζουμε ότι ένα honeypot τρέχει στη θύρα 22 του συστήματός μας, θα μπορούσαμε να ξεκινήσουμε μια συνεδρία απορρίπτοντας όλα τα πακέτα που προορίζονται για άλλες θύρες.

Τα φίλτρα που εφαρμόζονται μετά τη καταγραφή χρησιμοποιούνται πιο συχνά. Τα εν λόγω φίλτρα εισάγονται απευθείας στο κεντρικό παράθυρο του Wireshark και συγκεκριμένα στο πεδίο Filter, πάνω από τη «Λίστα Πακέτων». Νομίζουμε όμως ότι κάπου εδώ θα πρέπει να πούμε δυο λόγια για το πώς συντάσσονται τα φίλτρα. Όπως είπαμε παραπάνω, πρόκειται για λογικές εκφράσεις που αναφέρονται στα διάφορα χαρακτηριστικά των δικτυακών πακέτων. Επομένως, μπορούμε να χρησιμοποιήσουμε διάφορους λογικούς τελεστές, όπως το λογικό ΚΑΙ, το λογικό Ή κ.ά., καθώς και συγκριτικούς όπως το μεγαλύτερο από, το μικρότερο από κ.ο.κ. Στους πίνακες που ακολουθούν φαίνονται οι διαθέσιμοι τελεστές.

Περιγραφή	Τελεστής
Ίσο	==
Διάφορο	!=
Μεγαλύτερο	>
Μικρότερο	<
Μεγαλύτερο ή ίσο	>=
Μικρότερο ή ίσο	<=
ΚΑΙ: όλες οι εκφράσεις που ενώνονται πρέπει να είναι αληθείς	and
Ή: μία ή περισσότερες από τις εκφράσεις που ενώνονται ερκεί να είναι αληθής	or
Αποκλειστικό Ή (XOR): μία και μόνο μία από τις εκφράσεις που ενώνονται πρέπει να είναι αληθής	xor
ΟΧΙ: η έκφραση ή το σύνολο των εκφράσεων πρέπει να είναι ψευδής	not

Ας δούμε ορισμένα παραδείγματα. Έστω ότι θέλουμε να εμφανιστούν μόνο τα πακέτα που έχουν προέλευση τον υπολογιστή με διεύθυνση 192.168.1.114. Για να συντάξουμε ένα τέτοιο φίλτρο θα ξεκινήσουμε επιλέγοντας το κατάλληλο πρωτόκολλο. Εφόσον μιλάμε για μια διεύθυνση IP, ποιο πρωτόκολλο θέλουμε; Μα, φυσικά, το ίδιο το IP! Το φίλτρο μας, λοιπόν, θα πρέπει να έχει την ακόλουθη μορφή: ip.src == 192.168.1.114.



Για να αναλύσουμε την επίθεση ARP poisoning (κλασικά πράγματα) πρέπει να προσθέσουμε τις ακόλουθες δύο στήλες στη "Λίστα Πακέτων" του προγράμματος. Βλέπετε, έχουμε να κάνουμε με το ARP, το οποίο στο TCP/IP protocol stack βρίσκεται χαμηλότερα από το επίπεδο του IP!

Για το Wireshark, το πρωτόκολλο IP αποτελεί ένα αντικείμενο που περιέχει πολλά άλλα στοιχεία. Στο πρώτο τμήμα του φίλτρου επιλέγουμε το στοιχείο με όνομα src, του αντικειμένου με όνομα ip. Αυτό το στοιχείο αποτελεί το λεγόμενο source. Στη συνέχεια, η λογική έκφραση που αποτελεί το φίλτρο ταυτίζει αυτό το στοιχείο με την επιθυμητή διεύθυνση. Αυτό ήταν όλο!

Ένα δεύτερο παράδειγμα είναι το φιλτράρισμα για δικτυακή κίνηση στη θύρα 21 (FTP). Αφού ασχολούμαστε με θύρες, πρέπει να ξεκινήσουμε από το αντικείμενο που περιέχει το χαρακτηριστικό της θύρας. Δύο είναι τα πρωτόκολλα που επικοινωνούν χρησιμοποιώντας την έννοια των δικτυακών θυρών: το TCP και το UDP. Το FTP βασίζεται το πρώτο κι άρα μπορούμε να ξεκινήσουμε ως εξής: tcp.port == 21. Το φίλτρο μας μπορεί να θεωρηθεί έτοιμο. Ωστόσο, για λόγους πληρότητας μπορούμε να το επεκτείνουμε εξής: tcp.port == 21 or ftp. Για να ελέγξουμε την ύπαρξη ενός χαρακτηριστικού κάποιου πακέτου, αρκεί να το δηλώσουμε χωρίς να χρησιμοποιήσουμε κανέναν τελεστή.

Για να εφαρμόσουμε ένα φίλτρο που μόλις γράψαμε, πατάμε στο κουμπάκι Apply. Για να ακυρώσουμε την ισχύ κάποιου, πατάμε στο Clear. Hint: Πατώντας στο κουμπί Expression, εμφανίζεται ένα μεγάλο μενού επιλογών μέσα από το οποίο μπορούμε να κατασκευάσουμε τα επιμέρους τμήματα ενός φίλτρου. Βέβαια, με αυτόν τον τρόπο εμφανίζονται και τα μύρια όσα πρωτόκολλα υποστηρίζει το Wireshark, μαζί με τα στοιχεία του καθενός.

Graphing. Όσοι φίλοι έχουν προσπαθήσει να εντοπίσουν τα προβλήματα σε ένα δίκτυο, θα συμφωνήσουν ότι ένα ωραίο γράφημα της δικτυακής κίνησης αποτελεί σπουδαίο βοήθημα. Το Wireshark δεν θα μπορούσε να στερείται ενός τέτοιου εργαλείου. Έτσι, προσφέρει ορισμένες επιλογές για τη δημιουργία γραφημάτων, οι οποίες δείχνουν τη συμπεριφορά των δεδομένων κατά τη διάρκεια της καταγραφής. Για να δούμε τα βασικά γραφήματα που παράγει το Wireshark, αρκεί να επισκεφθούμε τη διαδρομή Statistics → IO Graph. Το παράθυρο που εμφανίζεται δείχνει στην ουσία μια γραφική αναπαράσταση της κίνησης που έχουμε καταγράψει. Εκεί μπορούμε να προσθέσουμε μέχρι και πέντε διαφορετικά φίλτρα, ώστε να σχηματιστούν ισάριθμα διαγράμματα!

Παράδειγμα ανάλυσης πακέτων

Νομίζουμε ότι μετά από όλη αυτή τη θεωρία, οφείλουμε να εξετάσουμε κι ένα παράδειγμα. Πριν προχωρήσουμε, σημειώστε ότι στην ιστοσελίδα του Wireshark (όπως και σε άλλους δικτυακούς τόπους) προσφέρονται για κατέβασμα αρκετά αρχεία pcap. Αυτά τα αρχεία καταγραφής είναι χρήσιμα για τη διεξαγωγή δοκιμών και κυρίως για εκπαιδευτικούς λόγους. Καλό θα ήταν να κατεβάσετε μερικά και να παίξετε μαζί τους, ώστε να ακονίσετε τις ικανότητες σας. Εμείς θα εξετάσουμε ένα κλασικό παράδειγμα επίθεσης, στην οποία έχουμε αναφερθεί αρκετές φορές: Το περίφημο ARP poisoning! Για περισσότερες πληροφορίες ανατρέξτε στο πέμπτο επεισόδιο της πρώτης σεζόν του deltaCast: `http://deltahacker.gr/deltacast-s01e05`. Για να ακολουθήσετε με επιτυχία τα βήματα που περιγράφονται παρακάτω, πρέπει να κατεβάσετε το αρχείο pcap με το οποίο εργαστήκαμε. Θα το βρείτε εδώ: `http://bit.ly/dh020arppoison`.

Εκ πρώτης, ανοίγοντας το αρχείο pcap με το Wireshark δεν βλέπουμε τίποτα που να μας προκαλεί εντύπωση. Με μια πρώτη ματιά, συμπεραίνουμε ότι κάποιος σερφάρει στο web από το ένα site στο άλλο (συναντάμε πολλά πακέτα HTTP και διάφορα DNS queries). Η αλήθεια είναι ότι η έρευνά μας δεν θα προχωρούσε πολύ, αν αγνοούσαμε ότι το πρωτόκολλο ARP λειτουργεί πιο χαμηλά από το IP και παίζει με τις διευθύνσεις MAC (Σ.τ.Ε. Βλ. σχετικό άρθρο στο deltaHacker 004, περί του TCP/IP stack). Έτσι, το επόμενο βήμα μας είναι να εμπλουτίσουμε τη “Λίστα Πακέτων”. Αυτό γίνεται ακολουθώντας τη διαδρομή Edit → Preferences → User Interface (αριστερά) → Columns. Στο παράθυρο που εμφανίζεται βλέπουμε τις στήλες που διαθέτουν οι γραμμές της “Λίστας Πακέτων”. Με άλλα λόγια, βλέπουμε τα στοιχεία που εμφανίζονται για κάθε πακέτο. Αρχικά, πατάμε το κουμπί Add και κάνοντας κλικ στα πεδία title και field type, δίνουμε αντίστοιχα τα εξής: MAC Src και Hw src addr (resolved). Στη συνέχεια πατάμε στο OK κι επαναλαμβάνουμε την ίδια ακριβώς διαδικασία. Αυτή τη φορά, όμως, δίνουμε τα εξής: MAC Dest και Hw dest addr (resolved). Είμαστε έτοιμοι! Επιστρέφοντας στο κεντρικό παράθυρο του Wireshark, θα δούμε ότι οι γραμμές στη “Λίστα Πακέτων” περιλαμβάνουν δύο νέες στήλες. Η μία δείχνει τη διεύθυνση MAC της πηγής του εκάστοτε πακέτου και η άλλη δείχνει τη διεύθυνση MAC του προορισμού.

Μελετώντας το αρχείο, παρατηρούμε ότι στο 1/3 του μήκους του (συγκεκριμένα στο πακέτο με αριθμό 54) συντελείται μια ενδιαφέρουσα αλλαγή. Μέχρι εκείνο το σημείο, ένας υπολογιστής με κάρτα Dell επικοινωνούσε με έναν Cisco router. Εκεί όμως συναντάμε τρία πακέτα του πρωτοκόλλου ARP και στη συνέχεια βλέπουμε ότι ο υπολογιστής με την κάρτα Dell αρχίζει να μιλάει με ένα άλλο σύστημα. Σίγουρα κάτι δεν πάει καλά εδώ! Αναρωτιέστε πώς το καταλαβαίνουμε; Από την αρχική κίνηση

(πριν το πακέτο 54) μπορούμε να εξάγουμε πως έχουμε δύο συσκευές με τα ακόλουθα στοιχεία:

Συσκευή	IP	MAC
Dell	172.16.0.107	00:21:70:c0:56:f0
Cisco	172.16.0.1	00:26:0b:21:07:33

Ξαφνικά (πακέτο 54), βλέπουμε μια νέα συσκευή (Hewlett-Packard) να στέλνει ένα ARP request, ρωτώντας ποιος υπολογιστής έχει τη διεύθυνση 172.16.0.107 (η διεύθυνση του Dell). Εδώ χτυπάει καμπανάκι: Αυτά τα πακέτα στέλνονται συνήθως στη διεύθυνση broadcast του δικτύου, ώστε να τα λάβουν όλες οι συσκευές και να απαντήσει η ενδιαφερόμενη. Στην περίπτωσή μας, όμως, το σχετικό ερώτημα στάλθηκε απευθείας στον υπολογιστή Dell. Ακολούθως (πακέτο 55), το θύμα απαντάει στον επιτιθέμενο στέλνοντας τη διεύθυνση MAC. Αμέσως μετά έρχεται το χτύπημα: Ο επιτιθέμενος, έχοντας τα στοιχεία που χρειάζεται, στέλνει μια απάντηση ARP στο θύμα (πακέτο 56), λέγοντας πως ο router (172.16.0.1) έχει πλέον άλλη διεύθυνση MAC. Αναρωτιέστε ποια διεύθυνση; Μα, τη δική του φυσικά: 00:25:b3:bf:91:ee. Ένα παρόμοιο πακέτο αποστέλλεται και στον router, αλλά αυτό δεν το βλέπουμε αφού η καταγραφή μοιάζει να έγινε από τον υπολογιστή του θύματος. Ο επιτιθέμενος λέει στον router ότι το IP 172.16.0.107 (ο υπολογιστής του θύματος) αντιστοιχεί σε μια νέα διεύθυνση MAC και δίνει πάλι τη δική του. Με αυτές τις δύο απάτες, ο επιτιθέμενος καταφέρνει να μπει ανάμεσα στο θύμα και στον router. Πλέον, το θύμα δεν επικοινωνεί απευθείας με τον router, αλλά οικειοθελώς στέλνει τα πακέτα του στον επιτιθέμενο. Το ίδιο συμβαίνει και με τον router, ο οποίος προωθεί τα πακέτα που προορίζονταν για το θύμα στον επιτιθέμενο... Τρομάξατε; Μην αγχώνεστε, αφού υπάρχουν αρκετές λύσεις και αντίμετρα για τις επιθέσεις του είδους. Περισσότερα θα βρείτε σε σχετικό άρθρο, στο site του περιοδικού: http://deltahacker.gr/?p=1583.

Επίλογος

Σε αυτό το άρθρο της σειράς μας, κάναμε μια βόλτα στα δικτυακά μονοπάτια με τη βοήθεια του Wireshark. Πρόκειται για ένα καταπληκτικό εργαλείο που λύνει τα χέρια κάθε αναλυτή και διαχειριστή δικτύων. Ωστόσο, μην περιμένετε ότι η πλήρης εκμάθησή του αποτελεί παιχνιδάκι. Αλλά, ευτυχώς, με εξάσκηση δεν υπάρχει τίποτα που να μη μαθαίνεται και, φυσικά, το Wireshark δεν αποτελεί εξαίρεση! Ελπίζουμε να ευχαριστιέστε τα άρθρα αυτής της σειράς, όσο κι εμείς κατά τη συγγραφή τους. Στο μέλλον θα συνεχίσουμε να ασχολούμαστε με την ανάλυση πακέτων, σε αυτή ή σε κάποια άλλη σειρά άρθρων. Σε κάθε περίπτωση, να είστε σίγουροι ότι θα συναντήσουμε το Wireshark ξανά ;)



Skill: Advanced

Tags: WOL, BIOS, NIC, broadcast, MAC address, magic packet, Python

Το ξύπνημα των μηχανών!

by multiPetros

Όλοι οι υπολογιστές, από τους πιο παλιούς έως και τους πιο σύγχρονους, ξεκινούν τη λειτουργία τους με κάποιον διακόπτη. Σιγά το νέο, θα μας πείτε, αλλά αμέσως θα σας πούμε να περιμένετε λίγο. Σε μερικές περιπτώσεις, που λέτε, το πάτημα ενός κουμπιού είναι από δύσκολο έως εντελώς αδύνατο -- αν μη τι άλλο λόγω της φυσικής μας απόστασης από τον υπολογιστή.

Εδώ κι αρκετά χρόνια η IBM έχει υλοποιήσει μια τεχνολογία που επιτρέπει να ενεργοποιούμε έναν υπολογιστή απομακρυσμένα. Η εν λόγω τεχνολογία, αν και δεν είναι καθόλου διαδεδομένη στους κύκλους των οικιακών χρηστών, υποστηρίζεται πλέον απ' όλους τους σύγχρονους υπολογιστές. Σε αυτό το άρθρο, λοιπόν, θα γνωρίσουμε την τεχνολογία Wake On LAN (WOL) και θα μάθουμε να την αξιοποιούμε. Ειδικά γι' αυτό το τελευταίο, θα φτιάξουμε κι ένα πρόγραμμα σε Python μέσω του οποίου θα είμαστε σε θέση να ξυπνάμε οποιονδήποτε υπολογιστή του τοπικού μας δικτύου, χωρίς να πατάμε κανέναν διακόπτη τροφοδοσίας!

Ένα από τα παράπλευρα αποτελέσματα της επικράτησης των ευρυζωνικών συνδέσεων είναι ότι, πλέον, πρακτικά όλοι ασχολούμαστε με τα τοπικά δίκτυα. Βλέπετε, κάθε τέτοια σύνδεση συνοδεύεται συνήθως κι από έναν router. Έτσι, όλοι μας αποκτήσαμε έναν πάμφθηνο τρόπο για να στήσουμε ένα μικρό, οικιακό δίκτυο. Και κάπως έτσι οι παλιοί υπολογιστές ξεσκονίστηκαν, φρεσκαρίστηκαν κι απέκτησαν νέα ζωή κι αποστολή, μέσα σε κάποιο τοπικό δίκτυο. Βέβαια, το συνεχώς αυξανόμενο κόστος της ενέργειας κι ο θόρυβος που παράγουν τα παλιά μηχανήματα (κι όχι μόνο), δεν επιτρέπουν σε κανέναν λογικό άνθρωπο να αφήνει τους υπολογιστές του ενεργοποιημένους σε μόνιμη βάση. Ωστόσο, όσοι έχουν στήσει ένα δίκτυο στο σπίτι θέλουν συχνά να εργάζονται με παραπάνω από ένα μηχανήματα. Δεν θα ήταν όμορφα αν μπορούσαμε να ενεργοποιούμε όποιον υπολογιστή του δικτύου επιθυμούμε, από εκείνον στον οποίο βρισκόμαστε και χωρίς να τρέχουμε να πατάμε διακόπτες; Για παράδειγμα, σκεφτείτε την περίπτωση που θέλουμε να κρατήσουμε backup της δουλειάς μας, αλλά ο file server, το θορυβώδες παλιό μηχάνημα με τους πέντε σκληρούς δίσκους, είναι απενεργοποιημένος. Δεν θα ήταν μεγάλη ευκολία αν μπορούσαμε να τον ενεργοποιήσουμε χωρίς να πάμε στην αποθήκη, στο υπόγειο ή όπου αλλού τον έχουμε καταχωνιάσει;

Καλωσορίστε το WOL!

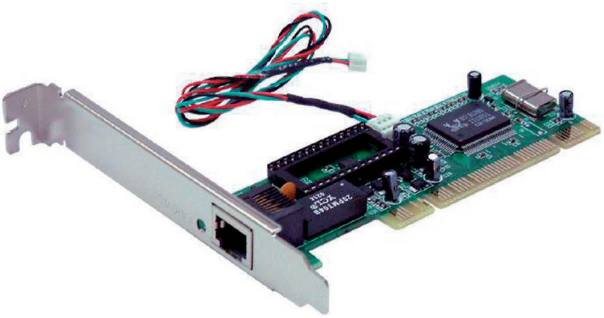
Το Wake on LAN αποτελεί μια τεχνολογία που επιτρέπει την ενεργοποίηση ενός υπολογιστή μέσω δικτύου. Κατά την απενεργοποίηση ενός υπολογιστή που υποστηρίζει το WOL, η κάρτα δικτύου πέφτει σε κατάσταση νάρκης κι αγνοεί όλα τα πακέτα που καταφτάνουν — εκτός από ένα *πολύ* συγκεκριμένο. Όταν ληφθεί αυτό το πακέτο, το κύκλωμα της κάρτας δίνει την κατάλληλη εντολή στη μητρική πλακέτα του υπολογιστή κι αυτός ξεκινά!

Πώς ενεργοποιείται;

Για τη λειτουργία αυτή του Wake On LAN, απαιτείται κατ' αρχάς η υποστήριξή του από το BIOS του συστήματος, όπως επίσης κι από την κάρτα δικτύου ή το κύκλωμα δικτύου της μητρικής. Ειδικά στην περίπτωση της κάρτας δικτύου, αν η έκδοση του διαύλου PCI της μητρικής είναι παλιότερη της 2.2[1], ενδέχεται να χρειαστεί ένα ειδικό καλώδιο. Αυτό το μικρό καλώδιο (έχει μόλις τρία pins) μεταφέρει το σήμα ενεργοποίησης του υπολογιστή και συνδέεται στην κάρτα δικτύου και σε ειδική υποδοχή της μητρικής. Οι νεότερες μητρικές (υποστήριξη PCI 2.2 και νεώτερο) υποστηρίζουν το λεγόμενο PME[2], το οποίο επιτρέπει την αποστολή του σήματος ενεργοποίησης μέσω του PCI bus, οπότε η ανάγκη για πρόσθετα καλώδια εξαλείφεται.

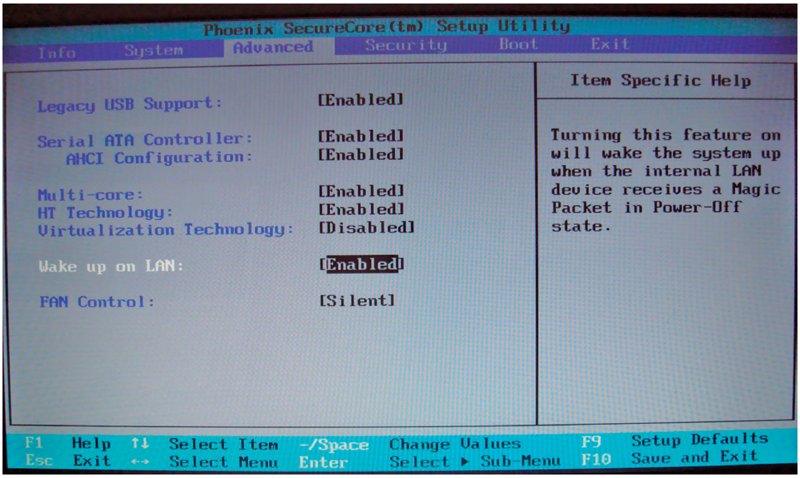
1. Αυτό συμβαίνει στις παλιές μητρικές
2. Power Management Events

Αφού ελέγξουμε την υποστήριξη του WOL από τη μητρική και κάνουμε την απαραίτητη σύνδεση (εφόσον αυτή απαιτείται), πρέπει να ξεκινήσουμε τον υπολογιστή και να μπούμε στο BIOS. Η σχετική ρύθμιση αναφέρεται ως Wake on LAN, Resume on LAN, Remote Wake-up ή Power-on by LAN. Για παν ενδεχόμενο μπορείτε να συμβουλευτείτε και το manual της μητρικής σας. (Σ.τ.Ε. <troll>Το οποίο σίγουρα έχετε κάπου προσεκτικά φυλαγμένο.</troll>)



Ορισμένες μητρικές που δεν υποστηρίζουν το PME απαιτούν τη χρήση ενός μικρού καλωδίου με 3pins, ώστε να λειτουργήσει το Wake On LAN.

Το Wake On LAN μπορεί να επαναφέρει τον υπολογιστή από οποιαδήποτε κατάσταση. Αυτές οι καταστάσεις είναι 6 (σύμφωνα με το πρότυπο ACPI[3]) και κωδικοποιούνται ως εξής: S1 - S5 για τις μορφές νάρκης, αδρανοποίησης, αναστολής κ.λπ. και G3 για την κατάσταση που ο υπολογιστής είναι πλήρως απενεργοποιημένος. Ωστόσο, ορισμένες φορές και ειδικά με την κατάσταση S1, παρατηρούνται μερικά προβλήματα και τότε θα πρέπει να ενεργοποιήσετε το Wake On LAN *και* από τις ρυθμίσεις του driver, μέσα στο λειτουργικό σύστημα.

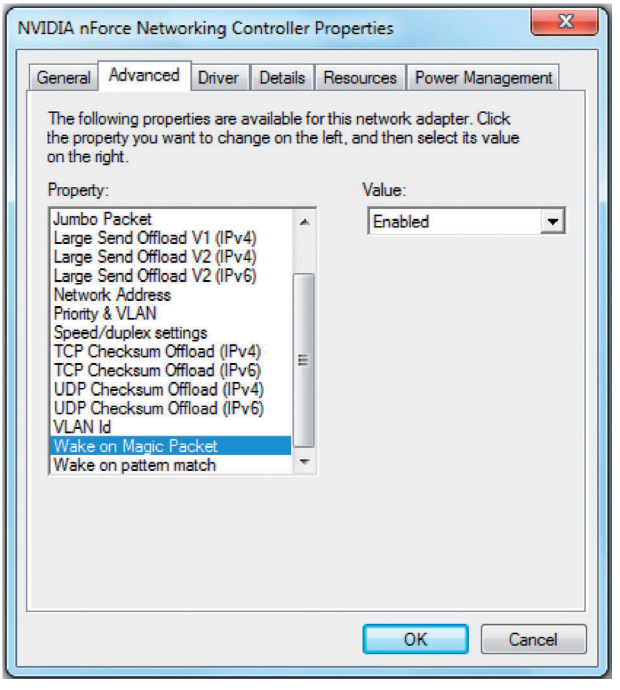


Για την ενεργοποίηση της λειτουργίας WOL είναι απαραίτητη η ενεργοποίησή της από το BIOS. Ανάλογα με το BIOS, η ορολογία μπορεί να διαφέρει με τα πιο συνηθισμένα ονόματα πέραν του Wake on LAN να είναι τα Resume on LAN, Remote Wake-up και Power-on by LAN. Για να βεβαιωθείτε, μπορείτε να συμβουλευτείτε και το εγχειρίδιο της μητρικής σας.

Το μαγικό πακέτο

Όπως αναφέραμε προηγουμένως, η κάρτα δικτύου δίνει σήμα για την εκκίνηση του υπολογιστή μονάχα όταν λάβει ένα πολύ συγκεκριμένο πακέτο. Αυτό το πακέτο είναι γνωστό ως Magic Packet και το περιεχόμενό του είναι κάθε φορά προσαρμοσμένο, ώστε να απευθύνεται σε μια συγκεκριμένη κάρτα δικτύου. Αυτή η διαφοροποίηση του Magic Packet είναι επιβεβλημένη, γιατί αν διαθέτουμε πολλούς υπολογιστές στο τοπικό μας δίκτυο κι έχουμε ενεργοποιήσει σε όλους το WOL, πρέπει να έχουμε τον έλεγχο του ποιος ακριβώς θα ξεκινά — αναλόγως βεβαίως της εντολής μας. Αν το Magic Packet είχε πάντα την ίδια μορφή, στο σενάριο που

3. Advanced Configuration and Power Interface



*Αν αντιμετωπίζετε προβλήματα με το WOL, ρίξτε μια ματιά στις ρυθμίσεις του driver της κάρτας δικτύου και φροντίστε να ενεργοποιήσετε *και* από εκεί το Wake On LAN!*

μόλις περιγράψαμε θα ξεκινούσαν όλοι οι υπολογιστές, ταυτόχρονα!

Οι κάρτες δικτύου (στο εξής θα τις αναφέρουμε κι ως NICs[4]) έχουν ένα μοναδικό αναγνωριστικό κωδικό, που ονομάζεται MAC Address[5] κι έχει μήκος 48bits. Αυτές οι διευθύνσεις αναπαρίστανται συνήθως με 6 ψηφία του δεκαεξαδικού συστήματος, τα οποία χωρίζονται σε ζευγάρια με άνω-κάτω τελείες ή με παύλες. Όπως υποψιάζεστε, το MAC Address ενσωματώνεται στα Magic Packets, ώστε τα εν λόγω πακέτα να είναι απόλυτα προσαρμοσμένα. Συγκεκριμένα, ένα Magic Packet αποτελείται από 102 bytes δεδομένων. Από αυτά, τα πρώτα 6 bytes έχουν πάντα την τιμή 255 (FF στο δεκαεξαδικό), ενώ τα υπόλοιπα 96 επαναλαμβάνουν 16 φορές το MAC Address της επιθυμητής κάρτας δικτύου (16 x 6 = 96). Το πακέτο της αφύπνισης αποστέλλεται ως ένα UDP datagram στη θύρα 7 (καμία σχέση με γνωστό club) ή στη θύρα 9, της broadcast address του τοπικού μας δικτύου. Έτσι, καταφτάνει σε όλες τις κάρτες του LAN. Αυτός ο τρόπος αποστολής αποτελεί μονόδρομο, αφού ένας απενεργοποιημένος υπολογιστής δεν διαθέτει διεύθυνση IP και το πακέτο δεν θα μπορούσε να σταλεί σε κάποια συγκεκριμένη διεύθυνση.

Περιορισμοί και προβλήματα

Το WOL υπόκειται σε κάποιους περιορισμούς, τους οποίους θα πρέπει να γνωρίζουμε για να αποφύγουμε τα προβλήματα ή/και τη δημιουργία λανθασμένων εντυπώσεων. Συγκεκριμένα, ισχύουν τα ακόλουθα:

- Η μετάδοση του Magic Packet μέσω UDP σημαίνει ότι δεν μπορούμε να επιβεβαιώνουμε τη λήψη του από την εκάστοτε κάρτα δικτύου.
- Οι ασύρματες κάρτες δικτύου απενεργοποιούνται πλήρως όταν το σύστημα μπαίνει σε κάποια κατάσταση χαμηλής ενέργειας κι ως εκ τούτου είναι αδύνατον να λάβουν magic packets.
- Το γεγονός ότι γίνεται broadcasting κι όχι routing του magic packet, καθιστά αδύνατη τη χρήση του WOL *εκτός* τοπικού δικτύου. Εκτός βέβαια κι αν ο router μας υποστηρίζει το λεγόμενο SDB (Subnet-Directed Broadcast): Με τη βοήθειά του ένας router μπορεί να λάβει κάποιο πακέτο, να το τροποποιήσει και στη συνέχεια να το εκπέμψει σε άλλο subnet. Ένας router που υποστηρίζει το SDB κι αφού τον έχουμε ρυθμίσει κατάλληλα, θα μπορούσε να λαμβάνει magic packets από το Internet και να τα εκπέμπει,

4. Network Interface Controllers
5. Media Access Control Address

μέσω UDP, στο τοπικό δίκτυο πίσω του. Μ' αυτόν τον τρόπο θα μπορούσαμε να αξιοποιούμε την τεχνολογία WOL ακόμα και όταν βρισκόμαστε εκτός τοπικού δικτύου. Ωστόσο, αυτή η ρύθμιση θα εξέθετε το τοπικό μας δίκτυο στον κίνδυνο να δεχτεί επιθέσεις DDoS (Smurf Attack).

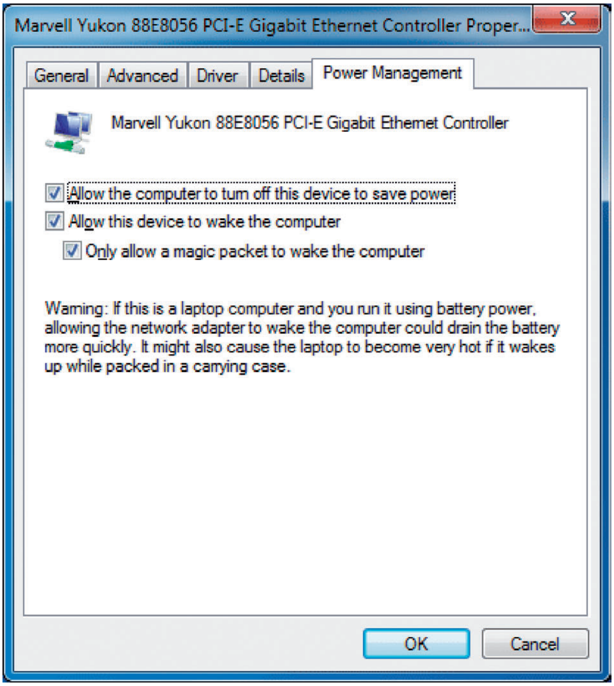
- Σε ορισμένες υλοποιήσεις, όταν ο υπολογιστής απενεργοποιηθεί απρόοπτα, π.χ., λόγω κάποιας διακοπής της παροχής του ρεύματος, δεν είναι δυνατή η εκ νέου ενεργοποίησή του μέσω WOL. Αν διαπιστώσουμε ότι ο υπολογιστής μας πάσχει απ' αυτή την ιδιοτροπία, μια λύση είναι να ρυθμίσουμε το BIOS ώστε μετά από διακοπές ρεύματος να επανέρχεται στην ενεργειακή κατάσταση που βρισκόταν προηγουμένως. Έτσι, μετά από απρόοπτες διακοπές της τροφοδοσίας ο υπολογιστής θα ενεργοποιείται ξανά, μόνος του.

Το λογισμικό

Μέχρι στιγμής αναφερθήκαμε στην τεχνολογία Wake On LAN, μιλήσαμε για το hardware που την υλοποιεί και περιγράψαμε το ρόλο και το περιεχόμενο των magic packets. Αυτό που δεν έχουμε πει είναι το πώς θα στείλουμε magic packets, ώστε να ξυπνήσουμε την ωραία κοιμωμένη. Ναι, σ' εκείνον τον προϊστορικό PIII αναφερόμαστε, με το άσπρο κουτί που με τα χρόνια έγινε μπεζ, τον οποίο έχουμε στριμώξει κάπου στο πατάρι. Όχι τίποτ' άλλο, αλλά θέλουμε να κατεβάσει την τελευταία έκδοση του openSUSE μέσω του αγαπημένου μας BitTorrent client, όση ώρα θα λείπουμε για καφέ ;)

Στους φορητούς υπολογιστές, τα Windows εξ ορισμού απενεργοποιούν την αφύπνιση μέσω της κάρτας δικτύου. Επομένως, αν έχετε laptop με Windows πρέπει να τσεκάρετε τις ρυθμίσεις Power Management και να ενεργοποιήσετε την επιλογή Allow this device to wake the computer.

Προγράμματα για αποστολή magic packets, αν και δεν είναι καθόλου διαδεδομένα, υπάρχουν αρκετά και μάλιστα για κάθε λειτουργικό (Linux, OS X, Windows και διάφορες πλατφόρμες κινητών). Επιπρόσθετα, όσοι έχετε εγκαταστήσει το DD-WRT ή το openWRT στον router σας, μπορείτε να στείλετε magic packets από το web interface του! Παρεμπιπτόντως, αν αυτά τα κάτι-WRT σάς είναι εντελώς άγνωστα, μπορείτε να ανατρέξετε στο τεύχος 011 για να μάθετε όλα τα σχετικά.



Εμείς βέβαια είμαστε κάπως ανήσυχα πνεύματα και οι έτοιμες λύσεις δεν μας αρέσουν ιδιαίτερα. Εξάλλου, θέλουμε το λογισμικό μας να είναι απόλυτα προσαρμοσμένο στις ανάγκες μας, ενώ δεν χάνουμε ποτέ την ευκαιρία να κατασκευάσουμε κάτι με τα χεράκια μας. Γι' αυτούς τους λόγους θα φτιάξουμε ένα δικό μας πρόγραμμα, για την αποστολή magic packets! Γνωρίζοντας όλη την απαραίτητη θεωρία, αρκεί να επιλέξουμε μια γλώσσα προγραμματισμού και να ξεκινήσουμε. Εμείς προτιμήσαμε την Python, επειδή επιτρέπει τη συγγραφή όμορφου και καθαρού κώδι-

κα, ενώ θα λειτουργεί και σε όλα τα λειτουργικά συστήματα. Για τις ανάγκες του παρόντος, εργαστήκαμε σε Windows. Παρακάτω θα εξετάσουμε τμηματικά το πρόγραμμα που γράψαμε.

```
import socket
import re
import argparse
```

Αρχικά, εισάγουμε τις απαραίτητες βιβλιοθήκες. Η socket παρέχει τις δυνατότητες δικτύωσης, η re διαθέτει τους μηχανισμούς για το χειρισμό και την κατανόηση των regular expressions από το πρόγραμμά μας, ενώ η argparse παρέχει υποστήριξη παραμέτρων κατά την εκτέλεση του προγράμματος. Βλέπετε, το πρόγραμμα που φτιάξαμε θέλουμε να λειτουργεί όπως τα περισσότερα της γραμμής εντολών του Unix, κι αυτό σημαίνει ότι η συμπεριφορά του θα ρυθμίζεται με την εισαγωγή παραμέτρων κατά την εκτέλεσή του.

```
def wol(macAddress, broadcastAddress, port):
    soc = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    soc.setsockopt(socket.SOL_SOCKET, socket.SO_BROADCAST, 1)

    magicPacket = "\xff"*6 + GetMacHex(macAddress)*16
    sendBytes = soc.sendto(magicPacket, (broadcastAddress,
port))
    return sendBytes
```

Η συνάρτηση wol αποτελεί κατά κάποιον τρόπο τον πυρήνα του προγράμματός μας και είναι αυτή που δημιουργεί και στέλνει το magic packet. Δέχεται σαν παραμέτρους τη macAddress του NIC του μηχανήματος που θέλουμε να αφυπνίσουμε, την broadcastAddress του δικτύου μας και τη θύρα που θέλουμε να στείλουμε το πακέτο. Αρχικά, ο κώδικας δημιουργεί ένα αντικείμενο socket, το οποίο και αρχικοποιεί και ρυθμίζει κατάλληλα, ώστε να δημιουργηθεί ένα UDP datagram. Εν συνεχεία δημιουργείται το περιεχόμενο του πακέτου. Εδώ, οι ευέλικτοι τελεστές χειρισμού συμβολοσειρών της Python μάς λύνουν τα χέρια! Δημιουργούμε μια συμβολοσειρά δεκαεξαδικών, όπου αρχικά επαναλαμβάνεται 6 φορές ο χαρακτήρας FF («\xff»*6). Ακολούθως, μέσω της συνάρτησης getMacHex, παίρνουμε τη macAddress σε δεκαεξαδική μορφή και την προσθέτουμε στο τέλος της συμβολοσειράς 16 φορές (+getMacHex(macAddress)*16). Κατόπιν στέλνουμε το πακέτο, μέσω της μεθόδου sendto του αντικειμένου socket. Στην εν λόγω μέθοδο, πέρα από τη συμβολοσειρά με τα περιεχόμενα του πακέτου δίνουμε και ένα tuple, με τη broadcast address του δικτύου και τη θύρα. Η μέθοδος επιστρέφει τον αριθμό των bytes που στάλθηκαν, τον οποίο αποθηκεύουμε στη μεταβλητή sendBytes.

```
def getMacHex(macAddress, separator=":"):
    if(macAddress.find(separator) == -1):
        separator = "-"
    macHex = macAddress.lower().replace(separator, "")
    if(re.match("[0-9a-f]{12}$", macHex)):
        return macHex.decode("hex")
    else:
        raise Exception("MAC address is not valid")
```

Η συνάρτηση getMacHex έχει διττό ρόλο. Από τη μια πραγματοποιεί τη μετατροπή της MAC address σε δεκαεξαδική μορφή, από την άλλη εκτελεί έναν έλεγχο εγκυρότητας. Δέχεται σαν παραμέτρους τη MAC address στη μορφή που έχουμε συνηθίσει να τη βλέπουμε και, προαιρετικά, τον χαρακτήρα που χρησιμοποιείται ως διαχωριστικό. Αρχικά, γίνεται ένας

έλεγχος για το αν ο διαχωριστικός χαρακτήρας που δώσαμε («:» ή «-») υπάρχει μέσα στη MAC address. Αν δεν τον βρει τότε ορίζει, αυθαίρετα, ως διαχωριστικό το χαρακτήρα «-». Στη συνέχεια μετατρέπουμε το περιεχόμενο της macAddress σε πεζά, ώστε να διευκολύνουμε τον έλεγχο μέσω κανονικών εκφράσεων, ενώ αφαιρούμε και το διαχωριστικό χαρακτήρα. Αμέσως μετά, με τη βοήθεια των κατάλληλων κανονικών εκφράσεων, ελέγχουμε αν η MAC address έχει δοθεί στη σωστή μορφή. Αν δηλαδή αποτελείται από ψηφία του δεκαεξαδικού συστήματος κι αν έχει μήκος 12 χαρακτήρες. Αν είναι όλα καλά, αποκωδικοποιούμε τους χαρακτήρες από τη δεκαεξαδική μορφή με τη χρήση της decode(). Διαφορετικά δημιουργούμε μια εξαίρεση, στην οποία περιγράφουμε ότι η MAC address δεν είχε τη σωστή μορφή.

```
def main():
    parser = argparse.ArgumentParser(description="Send WOL
magic packet to the selected MAC destination", epilog="(c) 2013,
multiPetros for deltaHacker magazine")
    parser.add_argument("mac", help="The MAC address of the NIC
you want to wake-up with the magic packet. Digits could be
separated by \':\' or \'-\'", metavar="MAC_ADDRESS")
    parser.add_argument("-b", "—broadcast",
default="192.168.1.255", dest="broadcast", help="The network
broadcast address, in dotted format: www.xxx.yyy.zzz",
metavar="BROADCAST_ADDRESS")
    parser.add_argument("-p", "—port", default="9", dest="port",
help="The listening port usually 7 or 9. (By default, the program
use port 9)", metavar="PORT_NUM")
    args = parser.parse_args()
    try:
        port = int(args.port)
        if((port < 1) or(port > 65535)):
            raise Exception
    except Exception:
        print("Port must be a valid integer between 1-65535")
        return
    try:
        sendBytes = wol(args.mac, args.broadcast, port)
        print("Done! " + str(sendBytes) + " bytes send")
    except socket.gaierror:
        print(args.broadcast + " is not a valid broadcast
address")
    except Exception as err:
        print(err)
```

Η συνάρτηση main() δεν θέλει ιδιαίτερες συστάσεις. Είναι αυτή που υλοποιεί τη διεπαφή με τον χρήστη. Για να πετύχουμε τη γνώριμη λειτουργικότητα εργαλείων του Unix, χρησιμοποιούμε τη βιβλιοθήκη argaparse της Python. Αρχικά, δημιουργούμε ένα ArgumentParser. Κατόπιν του προσθέτουμε τις παραμέτρους που θέλουμε να λαμβάνει από το χρήστη. Η μέθοδος add_argument κάνει όλη τη βρόμικη δουλειά! Μπορούμε να δηλώσουμε το όνομα της παραμέτρου σε σύντομη και εκτενή μορφή, το αν μια παράμετρος είναι προαπαιτούμενη ή όχι (αν όχι ορίζουμε την default τιμή), το κείμενο βοήθειας για την παράμετρο καθώς και ένα όνομα για την παράμετρο (metavar), με το οποίο θα εμφανίζεται στα διάφορα μηνύματα και στις συμβουλές χρήσης που θα παρουσιάζει το πρόγραμμά μας. Μια στιγμή! Κείμενο βοήθειας; Συμβουλές χρήσης; Ποιος τα φτιάχνει όλα αυτά; Μα, φυσικά, το ίδιο το αντικείμενο ArgumentParser!

Όταν είπαμε ότι αναλαμβάνει *όλη* τη βρόμικη δουλειά, το εννοούσαμε! Μετά τη ρύθμιση του ArgumentParser, έρχεται η σειρά να καλέσουμε τη μέθοδο parse_args(), ώστε να πάρουμε τις παραμέτρους που έδωσε ο χρήστης. Το αντικείμενο που επιστρέφει αυτή η μέθοδος έχει ως ιδιότητες τα ονόματα που ορίσαμε κατά τη δήλωση των arguments. Για να αποκτήσουμε πρόσβαση σε οποιοδήποτε από αυτά, μπορούμε αργότερα να χρησιμοποιήσουμε τον τελεστή «.».

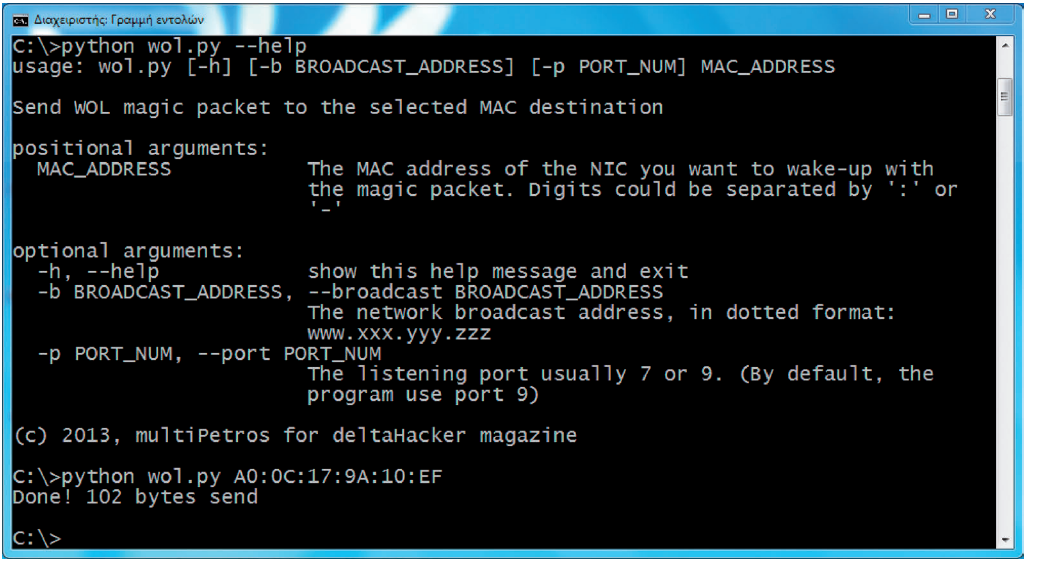
Εν συνεχεία μετατρέπουμε τον αριθμό της θύρας που έδωσε ο χρήστης (ή της προεπιλεγμένης τιμής που έχουμε ορίσει), από συμβολοσειρά σε ακέραιο αριθμό. Μετά κάνουμε κι έναν έλεγχο για το αν βρίσκεται εντός αποδεκτών ορίων (οι θύρες μπορούν να έχουν τιμή μεταξύ 1 και 65535). Σε οποιαδήποτε περίπτωση σφάλματος ή μη συμφωνίας με αυτόν τον περιορισμό, εμφανίζουμε μια προειδοποίηση και το πρόγραμμα τερματίζεται. Τέλος, καλείται η συνάρτηση wol κι αν πάνε όλα καλά, εμφανίζουμε ένα μήνυμα επιτυχίας μαζί με τον αριθμό των bytes που στάλθηκαν. Όχι πως έχει ιδιαίτερη χρησιμότητα κάτι τέτοιο, αλλά μιας και φτιάξαμε το δικό μας πρόγραμμα, είπαμε να το στολίσουμε και λίγο — με την καλή έννοια!

```
if __name__ == "__main__":
    main()
```

Για αυτό το τμήμα δεν νομίζουμε ότι χρειάζονται ιδιαίτερες εξηγήσεις! Πάντως, όσοι δεν το κατανοούν, μπορούν να ρίξουν μια ματιά στο τεύχος 006 και στο άρθρο του Sonic για την Python. Για τον τρόπο χρήσης του προγράμματος δεν θα σας πούμε τίποτα. Πιστεύουμε ότι έγινε προφανής από την περιγραφή του κώδικα. Στο κάτω-κάτω, μπορείτε να το τρέξετε χωρίς κάποια παράμετρο ή να δώσετε κάποια από τις γνωστές -h και —help, ώστε να λάβετε ένα σύντομο κείμενο βοήθειας.

Εμείς θα σας αφήσουμε για την ώρα, να παίξετε με το πρόγραμμα και φυσικά με τα μηχανήματα του τοπικού σας δικτύου. Τέλος, για να μην παιδεύεστε χωρίς λόγο, μπορείτε να κατεβάσετε το προγραμματάκι μας από το http://bit.ly/dh020wol.

Η έξοδος του προγράμματός μας. Αρχικά του ζητάμε πληροφορίες για τον τρόπο χρήσης του και στη συνέχεια στέλνουμε κι ένα magic packet!



Το **deltaHacker** είναι το ένα και **μοναδικό** περιοδικό με θεματολογία το ethical **hacking**, τα **δίκτυα** και την **ασφάλεια**, τον **προγραμματισμό** και τα **ηλεκτρονικά**.

Το **deltaHacker** δεν γυρνάει εδώ κι εκεί στα περίπτερα, αλλά για δεύτερη χρονιά στη σειρά **έρχεται κατευθείαν σ' εσάς**.

Αν είστε από εκείνους που δεν βλέπουν τον υπολογιστή τους ως άλλη μια οικιακή συσκευή και δεν αντιμετωπίζετε το Internet ως ένα δίκτυο που αρχίζει και τελειώνει με το Facebook και το Twitter, τότε είναι **109% βέβαιο ότι θα αγαπήσετε το deltaHacker**, καθώς κι ότι θα θέλετε την παρέα του **κάθε μήνα**.

Όλες οι πληροφορίες για όλα τα προγράμματα συνδρομών είναι στο

deltahacker.gr/subscriptions

Παραγγείλετε τώρα μία από τις συμφέρουσες συνδρομές από το

http://deltahacker.gr/pdf12order (ηλεκτρονική έκδοση)

ή από το

http://deltahacker.gr/order (έντυπη έκδοση, έως το τεύχος 019)

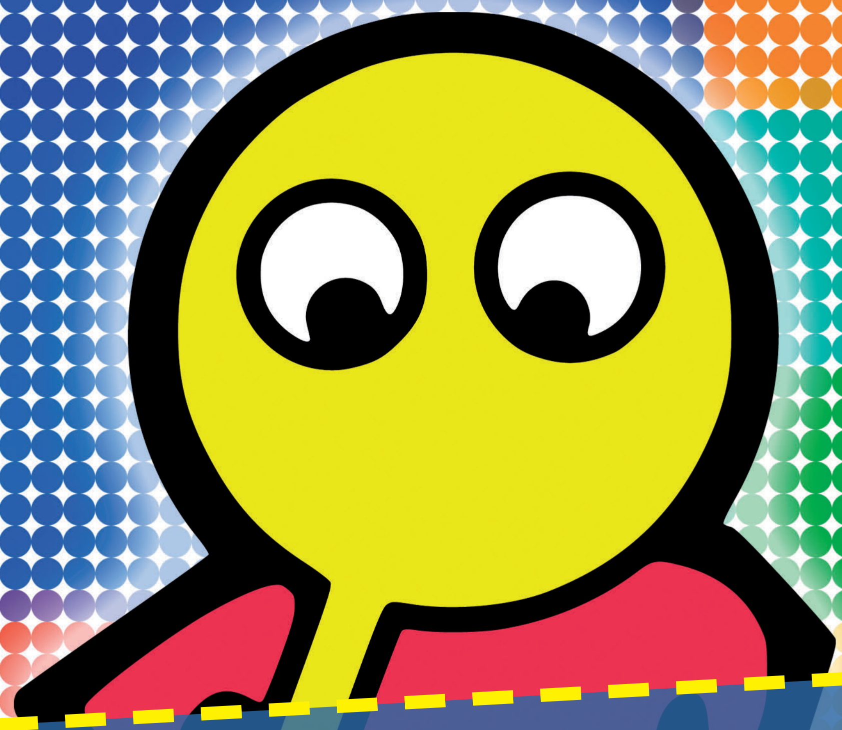
Σημείωση: Την ύλη **όλων των τευχών** που έχουν κυκλοφορήσει ως τώρα μπορείτε να τη δείτε στο

deltahacker.gr/category/magazine-issues

Skill: Beginner

Tags: I2P router, outproxy, FoxyProxy, susimail

Invisible Internet Project



Η ιδιαίτερη αγάπη μας για τα κρυπτογραφημένα δίκτυα (και τις πληροφορίες που μπορείς να βρεις σ' αυτά) ένα κακό έχει: δεν κρύβεται! Σε προηγούμενα άρθρα ασχοληθήκαμε με το Tor και το εξετάσαμε από όλες τις πιθανές σκοπιές. Αρχικά συνδεθήκαμε σ' αυτό ως απλοί χρήστες, ενώ στη συνέχεια συνεισφέραμε και στην επέκταση του δικτύου, λειτουργώντας ως κόμβοι. Το Tor όμως αποτελεί ένα μικρό κομμάτι αυτού που ονομάζουμε darknet -- και θα ήταν κρίμα αν περιοριζόμασταν σ' αυτό!

I2P

by GiannouG

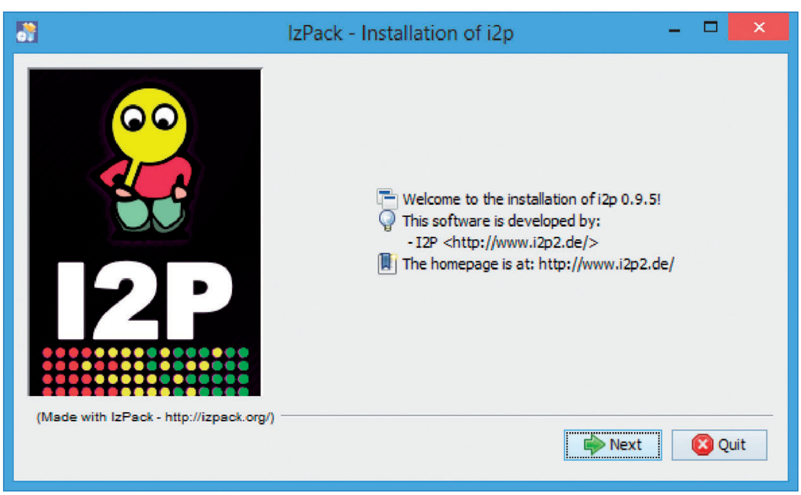
Υπάρχουν πολλά κρυπτογραφημένα δίκτυα που διαφέρουν σημαντικά μεταξύ τους, τόσο ως προς την υλοποίηση όσο και ως προς τη φιλοσοφία τους. Μην ανησυχείτε: Θα αποφύγουμε τις κουραστικές εισαγωγές όμως και ας αρχίσουμε τις συστάσεις. Αυτή τη φορά θα ασχοληθούμε με το Invisible Internet Project (IIP, I2P ή απλούστερα I2P). Το I2P υπόσχεται πλήρη ανωνυμία στις επικοινωνίες και στις συνδέσεις που γίνονται μέσα απ' αυτό, όπως ακριβώς συμβαίνει και με το Tor. Στο παρόν άρθρο θα δούμε πώς γίνεται η εγκατάσταση του σχετικού λογισμικού, θα ρυθμίσουμε *σωστά* τον Firefox και τέλος θα αναφέρουμε μερικές σελίδες απ' όπου μπορούμε να ξεκινήσουμε την περιπλάνησή μας σ' αυτόν το νέο κόσμο!

Το I2P προσφέρει ένα πέπλο ανωνυμίας πάνω από το Internet. Εκεί μέσα οι πληροφορίες μεταφέρονται ανώνυμα από κόμβο σε κόμβο.

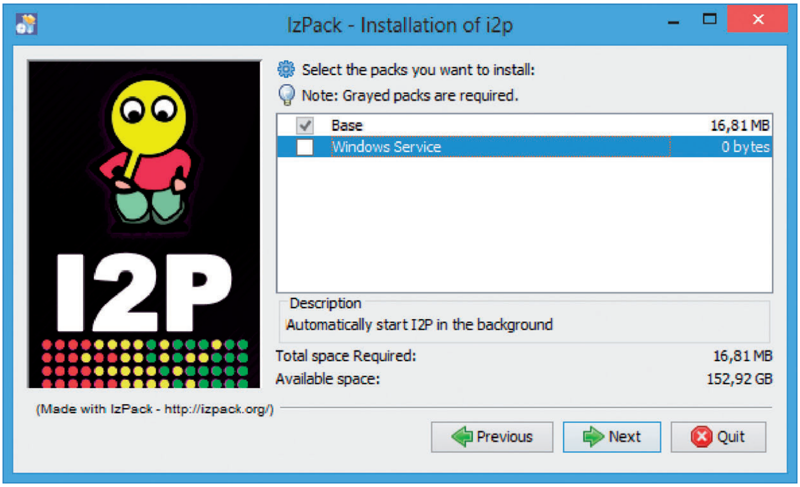
Tor εναντίον I2P

Όπως καταλαβαίνετε, η σύγκριση είναι αναπόφευκτη. Τι διαφορά έχει το Tor από το I2P και γιατί να επιλέξει κανείς το πρώτο απ' το δεύτερο ή το ανάποδο; Ας ξεκινήσουμε από τα βασικά. Το Tor δημιουργήθηκε για να προσφέρει *απόλυτη[1]* ανωνυμία στα μέλη του. Εντός του Tor δημιουργούνται *τυχαία* κυκλώματα και μέσω αυτών ο χρήστης βγαίνει στο απλό Internet (στην ορολογία του Tor ονομάζεται clearnet). Αντίθετα, το I2P δημιουργήθηκε με κύρια αποστολή να ενώσει όλους τους χρήστες σε ένα μεγάλο δίκτυο. Θα μπορούσαμε δηλαδή να φανταστούμε το I2P σαν ένα τεράστιο, κρυπτογραφημένο και ασφαλές LAN, όπου οι χρήστες του είναι ανώνυμοι και ανταλλάσσουν μεταξύ τους πληροφορίες. Αν θελήσουμε να βγούμε στο Internet μέσω του I2P πρέπει να βρούμε κάποιον κόμβο που να έχει ρυθμιστεί κατάλληλα και να επιτρέπει κάτι τέτοιο. Το I2P είναι ρυθμισμένο να βγαίνει στο Internet εξ ορισμού, αλλά στις δικές μας δοκιμές το πετύχαμε μοναχά μια φορά. Ο προεπιλεγμένος outproxy

Το I2P router είναι γραμμένο σε Java και τρέχει σε όλα τα λειτουργικά συστήματα. Η εγκατάστασή του σε Windows δεν απαιτεί τίποτα ιδιαίτερο.



1. Σε όλα τα *απόλυτα* υπάρχουν και μικρά γράμματα, όπως αυτή εδώ η πρόταση!



Καλό θα ήταν για αρχή να μην εγκαταστήσουμε το I2P ως Windows Service. Δεν θέλουμε να ανοίγει με τον υπολογιστή μας! Προτιμάμε να το ανοίγουμε χειροκίνητα.

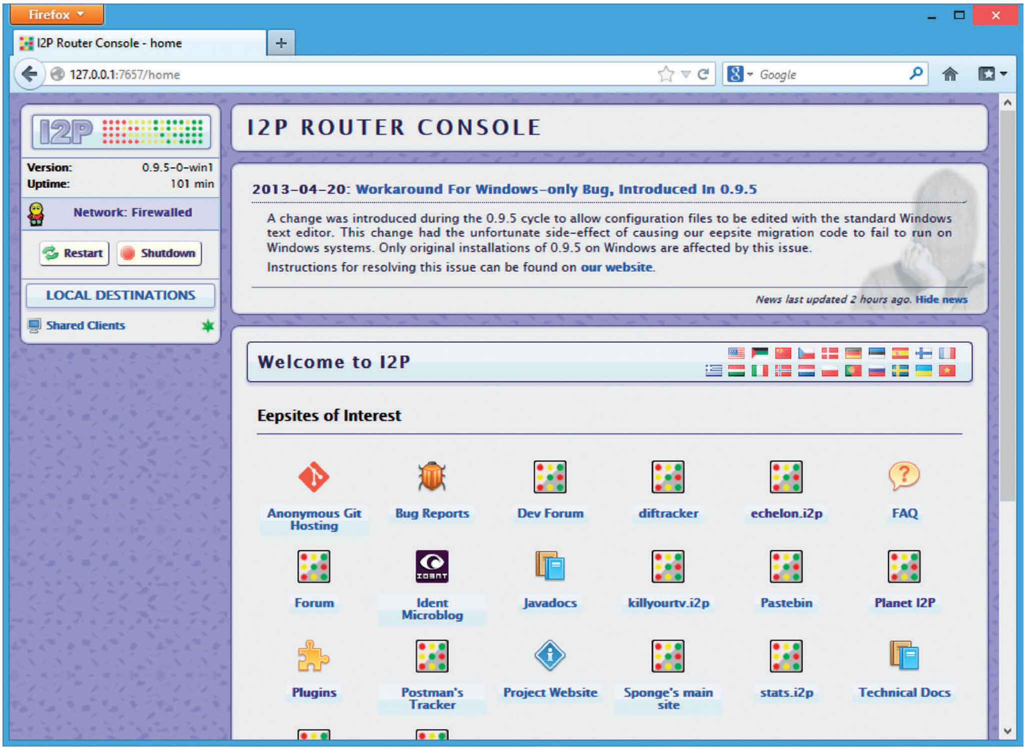
(ο κόμβος του I2P που οδηγεί στο Internet) χρησιμοποιεί το Tor. Δηλαδή, στην ουσία βγήκαμε στο Internet μέσω του Tor. Μπερδευτήκατε;

Σε πρακτικό επίπεδο, η λειτουργία των δύο διαφέρει σημαντικά. Μόλις ξεκινήσουμε το πρόγραμμα του I2P στο σύστημά μας, αυτό θα αρχίσει να αναμεταδίδει κρυπτογραφημένα πακέτα εντός του δικτύου, βοηθώντας στην επικοινωνία των υπόλοιπων χρηστών. Με άλλα λόγια, τα δεδομένα των χρηστών του θα περνάνε *και* από εμάς, χωρίς να έχουμε έλεγχο στο αν θα γίνεται αυτό ή όχι! Επιπλέον, η κεντρική βάση των χρηστών (NetDB) είναι διασκορπισμένη και δεν βρίσκεται κάπου κεντρικά, όπως γίνεται με το Tor. Θυμίζουμε ότι στο Tor, για να γίνουμε αναμεταδότες πληροφοριών (δηλαδή Tor relays), απαιτούνται ειδικές ρυθμίσεις. Στο I2P δεν μπορούμε να αφήσουμε τους υπόλοιπους χρήστες να βγαίνουν στο Internet από το δικό μας μηχάνημα. Τουλάχιστον όχι εύκολα. Για να πετύχουμε κάτι τέτοιο, πρέπει να φτιάξουμε μόνοι μας έναν κατάλληλο proxy server και το λογισμικό του I2P δεν θα μας βοηθήσει καθόλου σε αυτή τη δουλειά…

Εγκατάσταση

Το πρόγραμμα που μας ενδιαφέρει είναι το I2P router, η πύλη και ταυτόχρονα ο κόμβος μας στο δίκτυο I2P! Μπορούμε να κατεβάσουμε την κατάλληλη έκδοση για το λειτουργικό μας από την επίσημη ιστοσελίδα στο `http://www.i2p2.de/download.html`. Εδώ να σημειώσουμε πως δεν έχει σημασία σε ποια πλατφόρμα θα το εγκαταστήσουμε, μιας και είναι γραμμένο σε Java. Αρκεί δηλαδή το σύστημά μας να υποστηρίζει τη Java! Για το άρθρο χρησιμοποιήσαμε την τελευταία έκδοση των Windows, κυρίως για λόγους ευκολίας. Η εγκατάσταση είναι αρκετά απλή και ακολουθεί το μοτίβο του Next → Next → Finish ;) Αυτό που πρέπει να προσέξουμε είναι να απενεργοποιήσουμε την επιλογή Windows service, ώστε το I2P router να μην ξεκινάει αυτόματα μαζί με το σύστημά μας.

Μετά την εγκατάσταση, μπορούμε να το ξεκινήσουμε με τρεις διαφορετικούς τρόπους. Οι σχετικές συντομεύσεις βρίσκονται στο Start menu του συστήματος. Αν επιλέξαμε το I2P να εγκατασταθεί σαν service, μπορούμε να ανοίξουμε μια κονσόλα πατώντας το I2P router console. Η κονσόλα



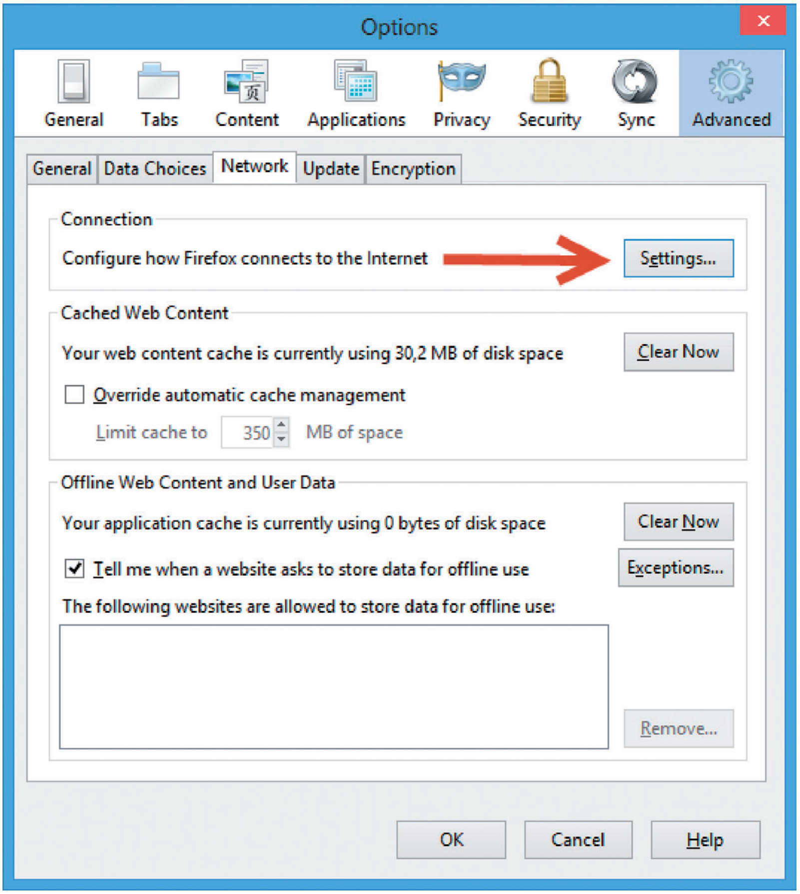
Το I2P router console είναι η πύλη μας προς το I2P. Εδώ βρίσκονται μερικοί σελιδοδείκτες προς γνωστές τοποθεσίες του I2P, ενώ παρέχονται κι όλες οι ρυθμίσεις του κόμβου μας!

είναι στην ουσία μια τοπική ιστοσελίδα[2], που προβάλει τις πληροφορίες του κόμβου μας. Με άλλα λόγια, η εν λόγω συντόμευση ανοίγει έναν browser (για κάποιο λόγο τον Internet Explorer) και φορτώνει τη διεύθυνση της κονσόλας (`http://127.0.0.1:7657/home`). Τώρα, αν δεν έχουμε εγκαταστήσει το I2P σαν υπηρεσία, μένουν οι υπόλοιπες δύο επιλογές. Η Start I2P (no window) και η Start I2P (restartable). Η πρώτη επιλογή ανοίγει έναν browser και το I2P router τρέχει στο background, ενώ η δεύτερη ανοίγει ένα παράθυρο εντολών (cmd) με τον router, καθώς κι έναν browser με την κονσόλα. Η διαφορά των δύο είναι ότι επιλέγοντας το restartable μπορούμε να παίξουμε με τον κόμβο και τις ρυθμίσεις του και να του κάνουμε επανεκκίνηση οποτεδήποτε επιθυμούμε, χωρίς να χρειάζεται να τον ξανατρέξουμε χειροκίνητα[3]. Για το άρθρο θα χρησιμοποιήσουμε την εκδοχή restartable.

Προετοιμασία Firefox

Όπως και στο Tor, έτσι και εδώ χρειαζόμαστε έναν browser μέσα από τον οποίο θα μπορούμε να περιηγούμαστε στο κρυπτογραφημένο δίκτυο. Για τη διαχείριση του I2P router χρησιμοποιείται εξ ορισμού ο Internet Explorer. Ωστόσο, για την περιήγηση στο δίκτυο I2P μπορούμε να χρησιμοποιήσουμε οποιονδήποτε browser. Αυτό που χρειάζεται να κάνουμε είναι να ορίσουμε τον κατάλληλο proxy server. Ο browser που θα βασι-

2. Οι προγραμματιστές πιθανότατα αποφάσισαν να γίνεται μέσω σελίδας η διαχείριση για να μην μπλέκουν με τα διαφορετικά GUI τις κάθε πλατφόρμας. Επιπλέον, μέσω HTTP η απομακρυσμένη διαχείριση είναι ευκολότερη!
3. Αυτή είναι και η δουλειά του παραθύρου CMD! Στην ουσία τρέχει κάποιο πρόγραμμα/wrapper που φροντίζει να ξεκινά και να σταματάει *αυτόματα* το I2P.

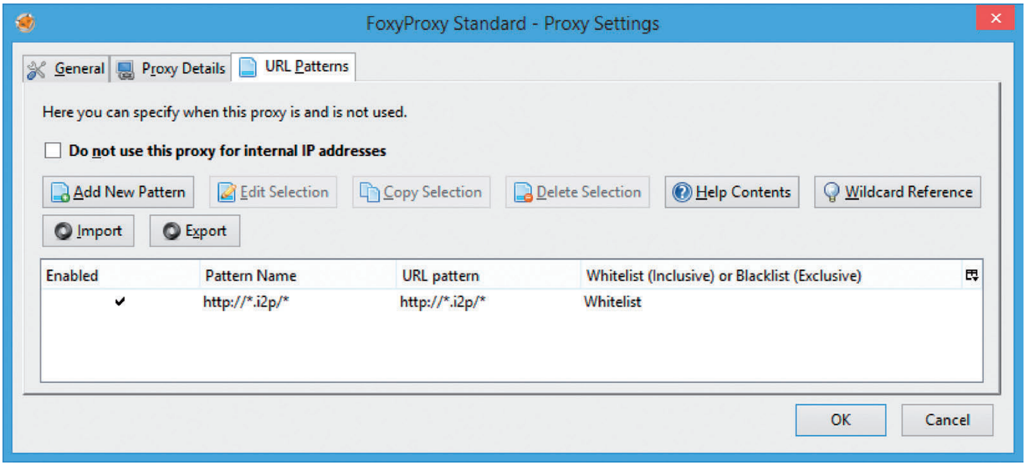


Οι ρυθμίσεις proxy του Firefox αντικαταστάθηκαν από αυτές του FoxyProxy. Πατώντας το κουμπί για τη ρύθμιση, εμφανίζεται το παράθυρο του plugin!

στούμε είναι ο Firefox κι αυτό γιατί χρειαζόμαστε ένα από τα plugins του[4]. Συγκεκριμένα, θα χρησιμοποιήσουμε το FoxyProxy για να ανακατευθύνουμε μόνο το domain του I2P (.i2p) μέσα από τον I2P router μας. Αν δεν μας ενδιαφέρει αυτή η λειτουργία ή αν θέλουμε να κάνουμε μια μικρή δοκιμή του δικτύου και μόνο, μπορούμε απλά να αλλάξουμε τον proxy που χρησιμοποιεί ο browser συνολικά. Σε αυτή την περίπτωση, αρκεί να ορίσουμε σαν HTTP proxy το 127.0.0.1 με το port 4444 και σαν HTTPS proxy το το 127.0.0.1 με το port 4445.

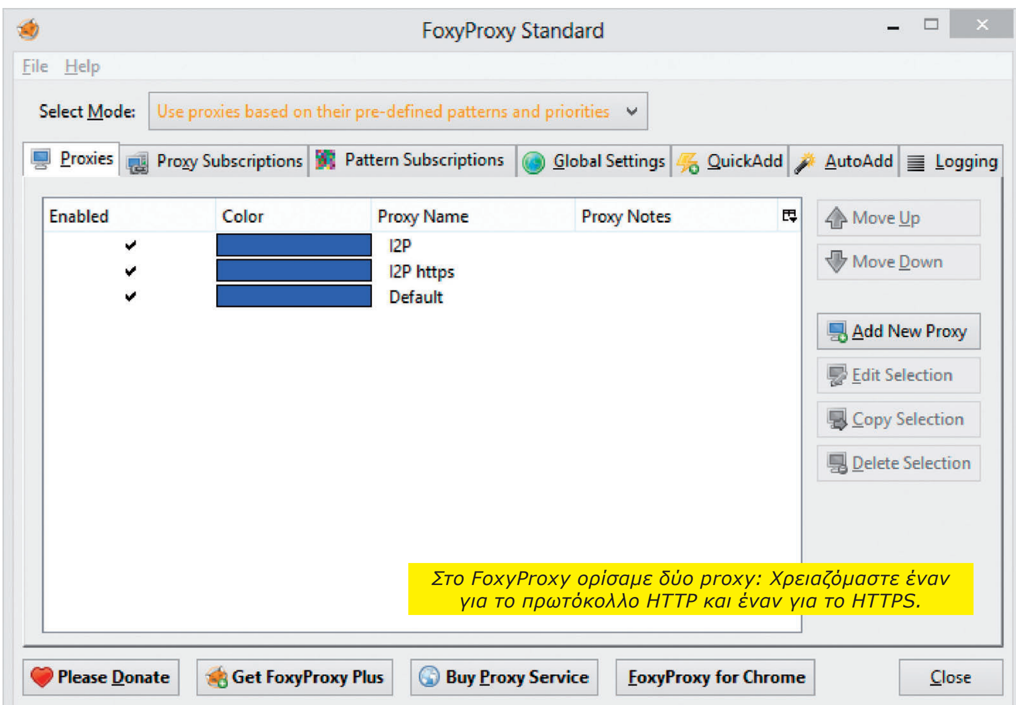
Αφού εγκαταστήσουμε το FoxyProxy Standard από το μενού των add-ons του Firefox (Firefox → Add-ons), το μενού ρύθμισης των proxy του browser θα αντικατασταθεί με αυτό του FoxyProxy. Εδώ μπορούμε να προσθέσουμε όσους κι όποιους proxy θέλουμε! Εμείς θα χρειαστεί να προσθέσουμε δύο επιλογές, για τα πρωτόκολλα HTTP και HTTPS. Οι σχετικές ρυθμίσεις γίνονται από το μενού ρυθμίσεων του προγράμματος κι από την περιοχή Advanced → Network → Settings. Αρχικά, κάνουμε κλικ στο Add new proxy. Στο παράθυρο που εμφανίζεται πατάμε το Manual Proxy Configuration και δίνουμε σαν IP το 127.0.0.1 και σαν port το 4444. Στη συνέχεια μεταβαίνουμε στην τρίτη καρτέλα, όπου προσθέτουμε ένα

4. Το συγκεκριμένο plugin υπάρχει *και* για Chrome *και* για Internet Explorer, αλλά ας μην του κλέψουμε την δόξα!

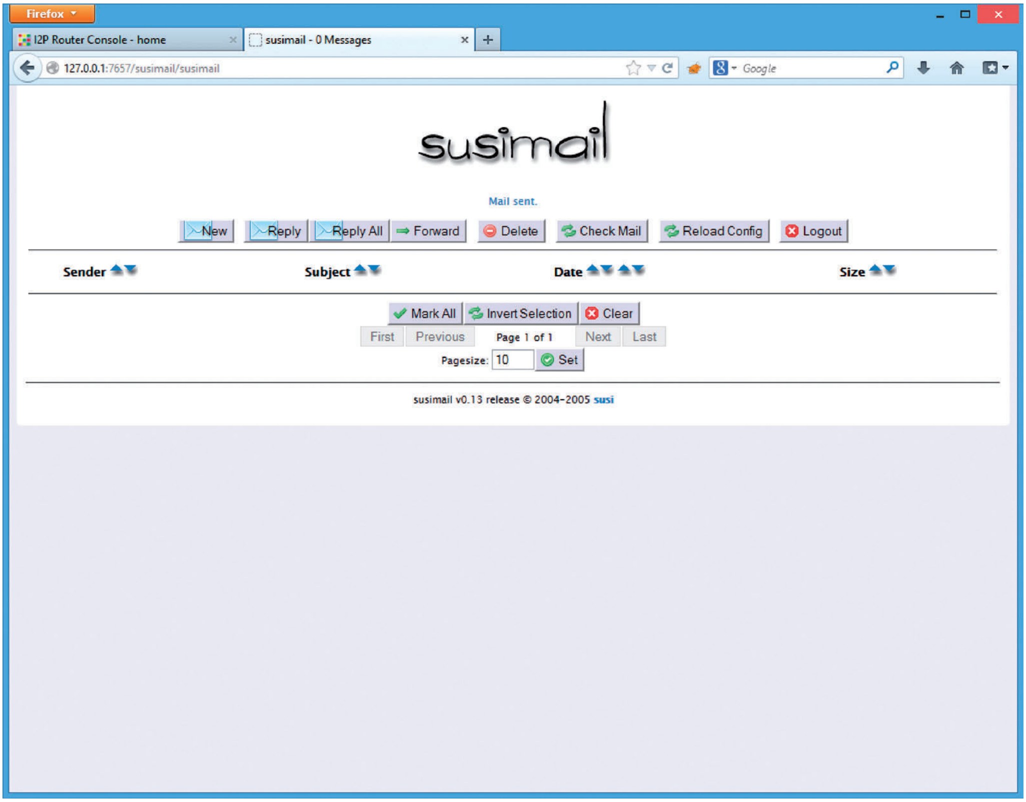


Με το FoxyProxy εξασφαλίσαμε ότι όλα τα requests προς τα domain που τελειώνουν σε «.i2p» θα δρομολογούνται μέσω του I2P router...

νέο pattern με το Add new pattern. Από εδώ ορίζουμε για ποια URL θα χρησιμοποιείται ο proxy που ορίσαμε. Το pattern που μας ενδιαφέρει είναι το http://*.i2p/*. Τώρα, γυρνάμε στην πρώτη καρτέλα και δίνουμε ένα όνομα σε αυτό το πακέτο ρυθμίσεων — για παράδειγμα «I2P». Επαναλαμβάνουμε ακριβώς τα ίδια βήματα για το πρωτόκολλο HTTPS κι αυτή τη φορά δίνουμε σαν port το 4445. Φυσικά, αλλάζουμε κατάλληλα και το pattern, το οποίο αυτή τη φορά γίνεται https://*.i2p/* (προστέθηκε ένα s μετά το http). Σαν όνομα για αυτό το πακέτο ρυθμίσεων δίνουμε και πάλι οτιδήποτε θέλουμε — για παράδειγμα «I2P (https)». Μετά από



Στο FoxyProxy ορίσαμε δύο proxy: Χρειαζόμαστε έναν για το πρωτόκολλο HTTP και έναν για το HTTPS.



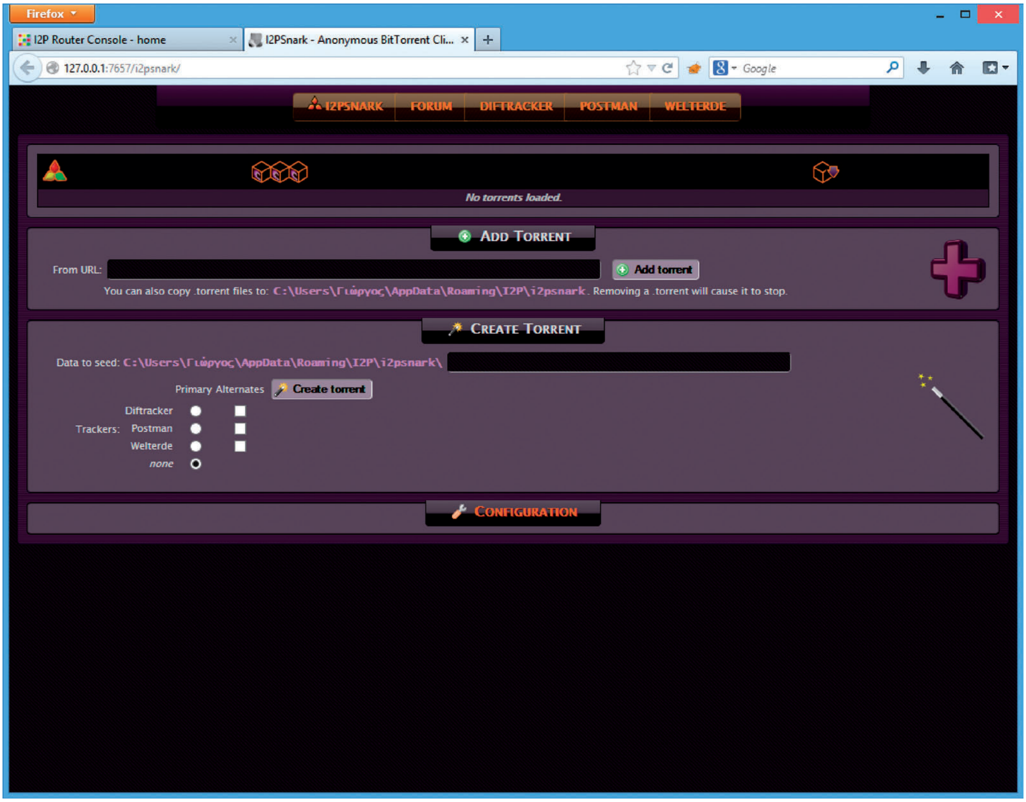
Σε συνδυασμό με το mail.i2p, το susimail αποτελεί την ιδανική λύση για όσους θέλουν να στέλνουν emails με πλήρη ανωνυμία.

αυτά τα βήματα είμαστε σχεδόν έτοιμοι. Τώρα πρέπει να σιγουρευτούμε ότι στο dropdown menu του FoxyProxy είναι ενεργοποιημένη η επιλογή «Use proxies based on their pre-defined patterns and priorities». Μόλις ξεμπερδέψουμε και με αυτό, ο Firefox θα ανακατευθύνει αυτόματα στον I2P router όλα τα αιτήματα προς τα domain που τελειώνουν σε «i2p»!

Τώρα, μπορούμε να ξεκινήσουμε τον I2P router κι αφού του δώσουμε μερικά λεπτά για να επικοινωνήσει με το υπόλοιπο δίκτυο, μπορούμε να τσεκάρουμε την κατάστασή του από την αριστερή sidebar της κονσόλας διαχείρισης.

Τα πρώτα βήματα

Η αρχική σελίδα του I2P router περιέχει bookmarks με μερικές από τις πιο διάσημες σελίδες του δικτύου. Από εδώ ξεκινούν συνήθως και οι περιπέτειές μας στο δίκτυο I2P! Σε αυτό το σημείο μπορούμε να κλείσουμε τον Internet Explorer και να ανοίξουμε την κονσόλα του I2P router στον Firefox. Για αρχή, καλό θα ήταν να εξερευνήσουμε λιγάκι τις ρυθμίσεις που προσφέρει ο router και να τροποποιήσουμε κάποιες, σύμφωνα με τα χαρακτηριστικά της σύνδεσης και του συστήματός μας. Συγκεκριμένα, καλό θα ήταν να αυξήσουμε λίγο τα όρια μεταφοράς δεδομένων. Από την αρχική σελίδα πατάμε το κουμπί Configure Bandwidth και εισάγουμε τις επιθυμητές τιμές σε kilobyte. Κατά τις δικές μας δοκιμές σε μια 24άρα γραμμή, δώσαμε 200kb/s download και 50kb/s upload.

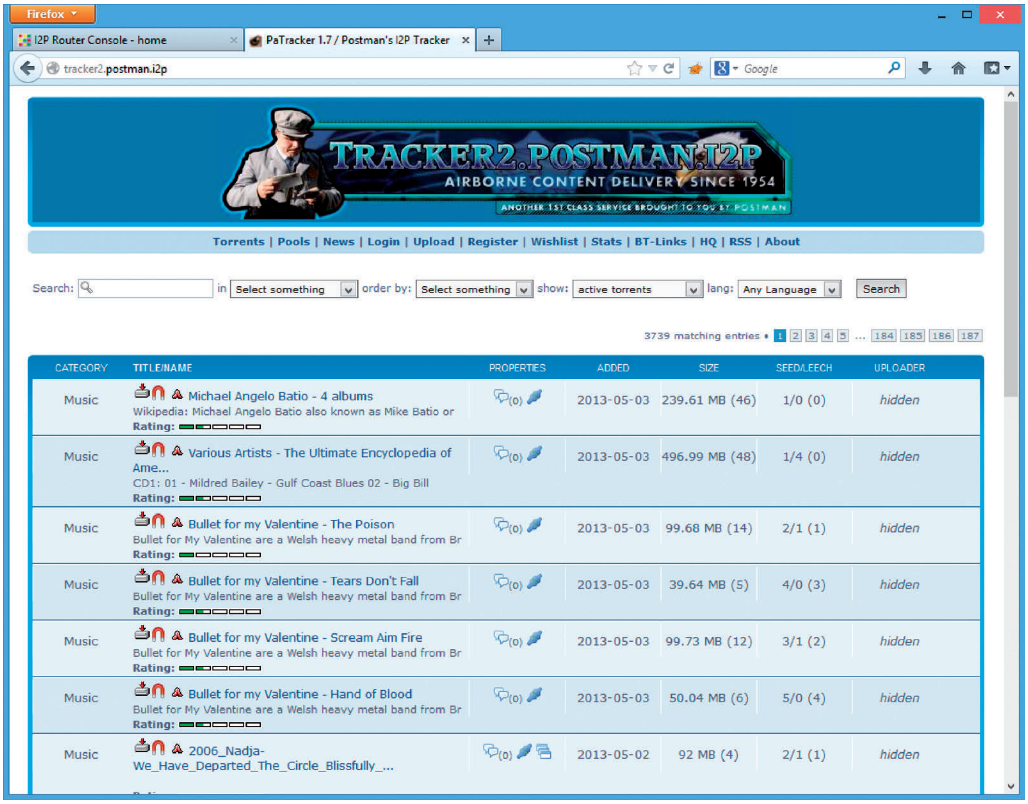


BitTorrent μέσα στο I2P; Μα φυσικά και ναι! Το μόνο που χρειαζόμαστε είναι (τουλάχιστον) ένα torrent file και τα υπόλοιπα θα τα αναλάβει το ενσωματωμένο πρόγραμμα...

Σε αντίθεση με το Tor, το I2P προσφέρει αρκετές προεγκατεστημένες υπηρεσίες *μαζί* με το I2P router. Από την αρχική σελίδα του κόμβου βλέπουμε ότι υπάρχουν προγράμματα για τη λήψη email αλλά και για το κατέβασμα torrent! Επιπρόσθετα είναι εγκατεστημένα και μερικά χρήσιμα προγράμματα, όπως το Addressbook και το I2PTunnel. Το πρώτο αποτελεί στην ουσία τον DNS server του I2P και διατηρεί ένα αρχείο παρόμοιο με το hosts file του συστήματός μας. Αυτό το αρχείο βοηθά στην αντικατάσταση των ακαταλαβίστικων διευθύνσεων του I2P σε απλά και κατανοητά ονόματα. Από την άλλη, το I2PTunnel επιτρέπει να ανοίγουμε θύρες επικοινωνίας από τον I2P router μας προς το υπόλοιπο δίκτυο. Για παράδειγμα, αν θελήσουμε να σηκώσουμε κάποιον FTP server θα χρειαστεί να τον δηλώσουμε εδώ και να δημιουργήσουμε το κατάλληλο tunnel. Αυτά τα ζητήματα όμως δεν θα μας απασχολήσουν για την ώρα. Η πιο ενδιαφέρουσα εφαρμογή που συνοδεύει τον I2p router είναι ο mail client. Συνδυάζοντας τον με την υπηρεσία `http://mail.i2p`, μπορούμε να στέλνουμε κρυπτογραφημένα emails εντός αλλά και εκτός του δικτύου. Το susimail, το πρόγραμμα για την λήψη και αποστολή των emails, βρίσκεται στα αριστερά της κονσόλας.

Προορισμοί με ενδιαφέρον

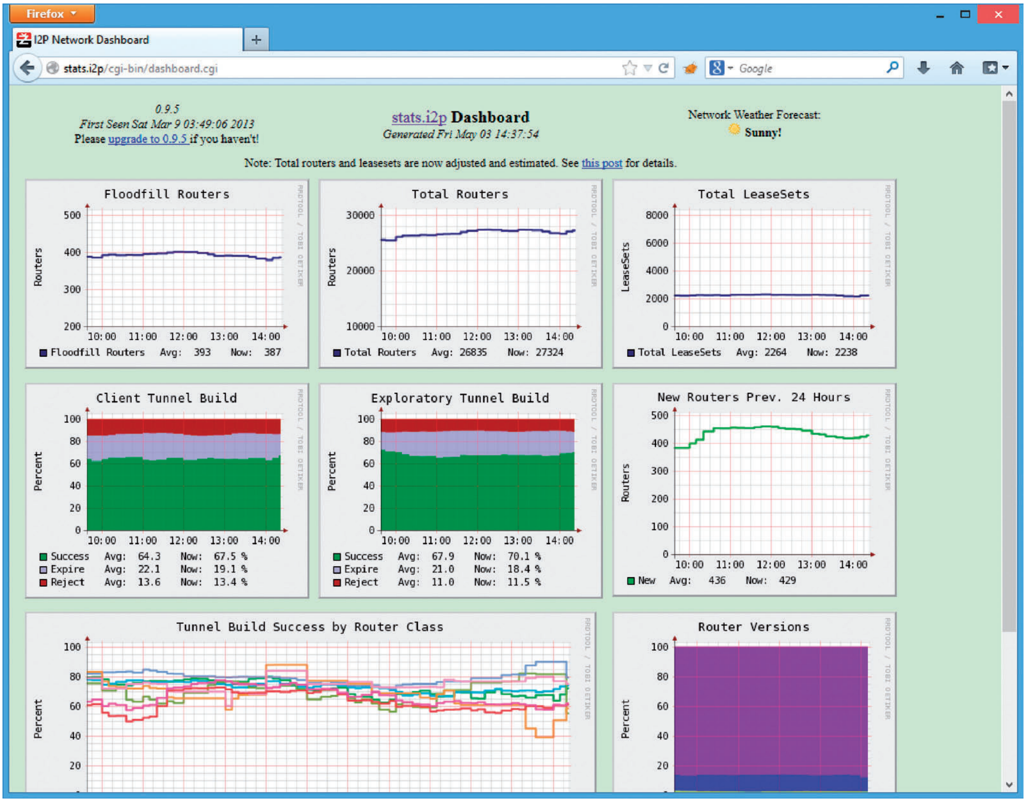
Δε θα το κρύψουμε, το I2P αποτελεί το βαρετό, δεκάχρονο ξαδερφάκι του Tor. Δεν θα βρούμε αρκετές σελίδες να εξερευνήσουμε και σίγουρα δε θα βρούμε δεκάδες «απόρρητα» έγγραφα για το πώς σκοπεύουν οι



Ο διασημότερος tracker εντός του I2P ονομάζεται postman's tracker (postman είναι ο διαχειριστής του mail.i2p) και φιλοξενεί αρκετές εκατοντάδες torrents με περιεχόμενο που θα ζήλευαν πολλοί trackers του απλού Internet!

εχθροί της Αμερικής να τη βομβαρδίσουν. Επίσης, δε θα βρούμε ούτε οδηγίες κατασκευής βόμβας. Για την ακρίβεια, δε θα βρούμε πολλά ενδιαφέροντα πράγματα εκτός από πληροφορίες και σελίδες που αφορούν στο I2P. Ελάχιστοι χρήστες ασχολούνται μ' αυτό και στην πλειονότητά τους είναι προγραμματιστές που απλά ενδιαφέρονται για την ανάπτυξή του. Ωστόσο, τα παραπάνω δεν σημαίνουν ότι είναι εντελώς αδιάφορο. Ένα από τα μεγαλύτερα και πιο ενδιαφέροντα site του I2P είναι το κεντρικό του forum. Μπορούμε να το επισκεφτούμε στο http://forum.i2p. Δεν απαιτείται εγγραφή για να διαβάσουμε, ούτε και για να δημοσιεύσουμε κάτι. Ακολουθώντας τον χαρακτήρα του δικτύου είναι κι αυτό εντελώς ανώνυμο. Μέσω του forum μπορούμε να ανακαλύψουμε πολλά πράγματα για το I2P, να συμβάλουμε στη λειτουργία αλλά και στην ανάπτυξή του, όπως επίσης να γνωρίσουμε κόσμο με παρόμοια ενδιαφέροντα.

Τα torrents δε θα μπορούσαν να λείπουν από το I2P. Εξάλλου, όπως είπαμε ήδη, ο I2P router συνοδεύεται εξορισμού από έναν torrent client! Ο postman's tracker είναι ο μεγαλύτερος torrent tracker σε ολόκληρο το I2P. Μπορούμε να τον επισκεφτούμε στο http://tracker2.postman.i2p. Με μια γρήγορη ματιά θα δούμε ότι δεν έχει τίποτα να ζηλέψει από τους γνωστούς tracker του απλού Internet. Η διαφορά όμως κρύβεται στο privacy και στο ότι πρακτικά ο καθένας μπορεί να ανεβάσει οτιδήποτε θέλει. Φανταστείτε το σαν ένα άδειο, σκοτεινό σοκάκι, στο οποίο μπαίνουν άγνωστοι και ανταλλάσσουν αρχεία. Σε αυτό το σοκάκι δεν υπάρχει κανένας έλεγχος για το τι διακινείται, οπότε τα κλικ μας θα πρέπει να



Σύμφωνα με το stats.i2p, αυτή τη συγκεκριμένη στιγμή υπάρχουν 27.324 ενεργοί I2P routers!

γίνονται με προσοχή! Ένας ακόμα γνωστός tracker βρίσκεται στο http://diftracker.i2p. Αν αντέχετε τα κακογραμμένα Αγγλικά, αξίζει να ρίξετε μια ματιά στα torrents που φιλοξενεί. Με την ίδια λογική λειτουργούν και τα δίκτυα Gnutella και eMule. Εμείς δεν παίξαμε καθόλου μαζί τους, αλλά όσοι ενδιαφέρονται μπορούν να βρουν τους αντίστοιχους clients στα http://www.imule.i2p και http://echelon.i2p/i2phex.

Μερικές σελίδες που κατάφεραν κι απορρόφησαν τον συντάκτη για αρκετές ώρες, είναι εκείνες ορισμένων developers ή και παλιών μελών του I2P. Η σελίδα με το μεγαλύτερο ενδιαφέρον και το περισσότερο υλικό βρίσκεται στη διεύθυνση http://echelon.i2p. Εκεί μπορούμε να βρούμε αρκετά προγράμματα για το I2P, οδηγούς αλλά και διάφορα νέα για τον κόσμο του I2P. Η σελίδα ανανεώνεται αρκετές φορές το μήνα και πρόσφατα ανέβηκε μια έκδοση του Bitcoin client, με υποστήριξη για I2P! Ξέρετε, για κρυπτογραφημένες μεταφορές εικονικών νομισμάτων μέσα από κρυπτογραφημένο δίκτυο[5]!

Ένας ακόμα ψαγμένος και ιδιαίτερα ανήσυχος χρήστης του I2P είναι ο killyourtv. Στη σελίδα του (http://killyourtv.i2p), εκτός από το προσωπικό του blog θα βρούμε αρκετές πληροφορίες για τα υπόλοιπα sites και τις υπηρεσίες που διατηρεί. Ενδιαφέρον έχει η έκδοση του Tahoe-LAFS, για το I2P. Πρόκειται για ένα σύστημα cloud εντός του I2P, στο οποίο κάθε χρήστης αναλαμβάνει να διατηρεί τόσο τα δικά του αρχεία, όσο κι εκείνα

5. *γκουχ* για ξέπλυμα μαύρου χρήματος *γκουχ*

των υπολοίπων χρηστών. Είναι σαν να λέμε ένα κοινωφελές σύστημα cloud, χωρίς κεντρική διοίκηση! Επιπρόσθετα, ο killyourtv είναι υπεύθυνος και συντηρεί έναν από τους μεγαλύτερους IRC servers μέσα στο I2P.

Τέλος, μια σελίδα που μας ενημερώνει για όσα συμβαίνουν μέσα στο I2P είναι το http://stats.i2p. Εκεί παρέχονται πληροφορίες για το πλήθος των συνδεδεμένων routers, για τις συνδέσεις μεταξύ αυτών, τους τύπους των συνδέσεων και πολλά άλλα. Όλα αυτά μπορεί να φαίνονται αδιάφορα για τους περισσότερους χρήστες, αλλά εμείς γουστάρουμε πολύ τα ωραιότατα σχεδιαγράμματα που διατηρούνται στη σελίδα :D

Κλείνοντας το δίκτυο

Είναι γεγονός ότι το Tor έχει περισσότερο υλικό. Ωστόσο, αν το καλοσκεφτεί κανείς, αυτό έχει να κάνει με τη δημοτικότητα του ενός ή του άλλου δικτύου. Όσο περισσότερα άτομα ασχολούνται με ένα δίκτυο, τόσο πιο ενδιαφέρον το κάνουν! Αυτό φυσικά δε σημαίνει ότι το I2P είναι άχρηστο. Φανταστείτε το Tor σαν μια πλατεία γεμάτη παιδάκια που τρέχουν ασταμάτητα πάνω-κάτω, φωνάζοντας κακές λέξεις. Το I2P αποτελεί ένα παγκάκι εντός αυτής της πλατείας, στο οποίο κάθονται ώριμοι άνθρωποι και συζητούν για το πώς θα κάνουν την πλατεία καλύτερη.



Το deltahacker.gr μέσα από το I2P! Είναι η μόνη σελίδα του κανονικού Internet που καταφέραμε να φορτώσουμε μέσα από το δίκτυο. Όπως βλέπετε η σελίδα φόρτωνε επ' άπειρον, γεγονός που δείχνει πόσο αργή είναι η σύνδεση του προεπιλεγμένου outproxy...

Skill: Intermediate

Tags: FreeBSD, dump, restore, ZFS

FreeBSD Cloning

the Right Way(TM)

Ζουν ανάμεσα μας. Όχι, δεν θα σας μιλήσουμε για ιστορίες συνωμοσίας και εξωγήινους. Για τους χρήστες Windows, λέμε. Κι όχι όλους, αλλά για μια συγκεκριμένη κατηγορία: τους Power Users! Ξέρετε, αυτούς που φτιάχνουν το PC τους μόνοι τους. Ποια είναι τα βασικά τους εργαλεία; Πού το πάμε μ' αυτό το άρθρο; Είναι ο τίτλος του ενδεικτικός;





by Sonic2000gr

Αλλά ας επιστρέψουμε στο πρώτο ερώρημα. Τι περιλαμβάνει η εργαλειοθήκη του Windows Power User;

- Το CD/DVD εγκατάστασης
- Το partition/CD/εξωτερικό σκληρό με τα προγράμματα και τους drivers
- Το πρόγραμμα Ghost

Ας μείνουμε λίγο σ' αυτό το τελευταίο. Είμαστε σίγουροι ότι όλοι το γνωρίζετε και πιθανόν το έχετε χρησιμοποιήσει συχνά τις παλιές, κακές εποχές, πριν βάλετε FreeBSD. Κι ακόμη και σήμερα το βλέπετε και το ακούτε συχνά: Ο καλός σας φίλος, ο Γιάννης (Σ.τ.Ε. The Proverbial Neighbor?!), έχει ένα μόνο ελάττωμα: Εξακολουθεί να πιστεύει στα λειτουργικά που βγαίνουν από το Redmond. Έρχεται στο σπίτι σας και σας διηγείται, μεσ' στην περηφάνια, πόσο γρήγορα κατάφερε να ολοκληρώσει την επανεγκατάσταση των Windows όταν άλλαξε μητρική (γιατί η προηγούμενη κάηκε), καθώς η παλιά του εγκατάσταση δεν εκκινούσε πλέον. Ή πόσο γρήγορα επανέφερε τον υπολογιστή του χρησιμοποιώντας το Ghost, όταν πριν δυο βδομάδες τον χτύπησε εκείνος ο τελευταίος ιος που, εντελώς απρόσεκτα κι απερίσκεπτα, ο μικρός αδερφός του κατέβασε μαζί μ' ένα άλλο πρόγραμμα.

Τώρα, τι ακριβώς δουλειά έχει ο μικρός αδερφός του Γιάννη να εκτελεί χρέη διαχειριστή σ' ένα μηχάνημα Windows, αυτό το γνωρίζει μόνο ο Γιάννης. Μπορεί ο Γιάννης να υπερηφανεύεται για τα κατορθώματα του πριν μια βδομάδα, ένα μήνα ή ένα εξάμηνο. Εσείς τον παρακολουθείτε, κουνάτε συγκαταβατικά το κεφάλι και χαμογελάτε. Δεν θέλετε να τον εκνευρίσετε: Στα χρονικά διαστήματα που σας περιγράφει, εσείς δεν έχετε κάνει ούτε ένα reboot στο FreeBSD σας. Στα πλαίσια της καλής φιλίας σας, όμως, η εντολή uptime έχει απαγορευτεί στα τερματικά που έχετε διάσπαρτα στο σπίτι σας.

Τα παραπάνω βέβαια εγείρουν και κάποια ερωτήματα, όπως:

- Καλά, δεν χαλάει ποτέ το FreeBSD;
- Πόσο χρήσιμο θα ήταν κάτι αντίστοιχο του Ghost για UNIXοειδή λειτουργικά;

Θα σας απαντήσω χωρίς περιστροφές: Ναι, το FreeBSD χαλάει! Το δικό μου έχει ξεβγάλει ήδη μία μητρική και δύο τροφοδοτικά. Εκτός δηλαδή κι αν αναφέρεστε στο λειτουργικό, όχι στο μηχάνημα. Αλλά, ναι, κι αυτό χαλάει, όμως με πολύ συγκεκριμένους τρόπους και συνήθως με το σύνδρομο ΔΑΛΕΚ.

Δεν εννοούμε τα Daleks, που κυνηγάνε τον Dr Who. Το σύνδρομο ΔΑΛΕΚ, γνωστό επίσης ως Διαχειριστική Ανεπάρκεια Λόγω Έλλειψης Καφεΐνης, εκδηλώνεται τυπικά τις πρώτες πρωινές ώρες *πριν* επιδράσει το ευεργετικό ρόφημα στον εγκέφαλο του root. Έτσι, π.χ., η εντολή rm -rf * εκτελείται σε εντελώς λάθος κατάλογο κι από εντελώς λάθος χρήστη. Αλλά ας μην τα λέμε αυτά, γιατί πονάνε. (Κι εδώ που τα λέμε, τι δουλειά έχετε πρωί πρωί στο root terminal; Φιλαράκια με τον αδελφό του Γιάννη, είστε; Ε;)

Όσον αφορά τη δεύτερη ερώτηση, ναι, είναι χρήσιμο να έχουμε έναν τρόπο να κλωνοποιήσουμε το FreeBSD μας. Δεν είναι απαραίτητο να χαλάσει κάτι για να μας φανεί χρήσιμη μια τέτοια λειτουργία. Δείτε μερικά σενάρια.

- Θέλετε να περάσετε το λειτουργικό στον καινούργιο, τιτανοτεράστιο δίσκο των 4TB που μόλις αποκτήσατε, καθώς τα 500GB που έχετε τώρα σάς φαίνονται να έχουν μέγεθος... boot sector! (Hint: Σε έξι μήνες το ίδιο θα σας φαίνεται κι ο δίσκος των 4TB.)
- Έχετε με κόπο κτίσει μια desktop εγκατάσταση FreeBSD από τα ports κάνοντας compile τα πάντα — ακόμη κι αυτό το τέρας, το LibreOffice. Το βράδυ, στις ειδήσεις, ακούσατε κάτι για ένα παγόβουνο που αποκολλήθηκε από την Ανταρκτική κι αναρωτηθήκατε αν είχατε κάποια ανάμειξη σ' αυτό. Θα προτιμούσατε, στο επόμενό σας FreeBSD box, να αντιγράψετε την εγκατάσταση που ήδη έχετε, αντί να τα κάνετε όλα από την αρχή και ν' ακούσετε πάλι στις ειδήσεις για περιπλανώμενα παγόβουνα.
- Έχετε μια πετυχημένη εγκατάσταση και θέλετε να φτιάξετε ένα ολόκληρο εργαστήριο με FreeBSD boxes, που να βασίζεται σε αυτή. Εξάλλου, ένα μηχάνημα FreeBSD δεν είναι ποτέ αρκετό.
- Τέλος, ίσως σας χτύπησε το ΔΑΛΕΚ και πρέπει να επανέλθετε γρήγορα σε καλή, λειτουργική κατάσταση.

Το cloning, λοιπόν, είναι χρήσιμο και σ' μας, τους FreeBSD (r, l)oyals. Αλλά μη μου ζητάτε πρόγραμμα αντίστοιχο του Ghost, γιατί θα σας μαλώσω. Αν και υπάρχουν τέτοια προγράμματα, δεν πρέπει να ξεχνάμε ότι ένα UNIX σύστημα δεν είναι ίδιας λογικής με ένα σύστημα Windows. Δεν χρειαζόμαστε τίποτα που δεν υπάρχει ήδη στο FreeBSD base system, για να μεταφέρουμε την εγκατάσταση ή/και τα δεδομένα μας σ' ένα νέο δίσκο!

Γιατί δεν χρειαζόμαστε κάτι αντίστοιχο του Ghost;

Την απάντηση θα τη βρείτε αν μελετήσετε τη διαδικασία εκκίνησης του FreeBSD (σας παραπέμπω κατευθείαν στο FreeBSD Handbook!):

- Το BIOS φορτώνει το boot sector. Εκεί, ένα μικρό πρόγραμμα φορτώνει ένα λίγο μεγαλύτερο κι αυτό ένα λίγο μεγαλύτερο, το οποίο ξέρει να διαβάζει —στοιχειωδώς— το filesystem μας (UFS ή ZFS).
- Το τελευταίο αυτό πρόγραμμα φορτώνει τον πυρήνα και τον εκτελεί.
- Ο πυρήνας ανιχνεύει τις συσκευές, προσαρτεί τα συστήματα αρχείων, ξεκινά τις υπηρεσίες και τις ρυθμίσεις — πάντα σύμφωνα με αυτά που διαβάζει στο rc.conf— και μας φτάνει ωραία και καλά μέχρι το login.

Ίσως το απλοποιήσαμε λίγο, αλλά τι θέλουμε να σας δείξουμε εδώ;

Δεν υπάρχουν μαγικά βήματα κατά την εκκίνηση ενός συστήματος FreeBSD. Μπορούμε να την ακολουθήσουμε από την αρχή μέχρι το τέλος. Ποτέ δεν θα δείτε FreeBSD με... μπλε οθόνη στην εκκίνηση. Για οποιονδήποτε λόγο κι αν σταματήσει, ανεπιτυχώς, μια εκκίνηση, πάντα θα μπορείτε να βρείτε τι φταίει και να το φτιάξετε. Στη χειρότερη περίπτωση (ΔΑΛΕΚ επιπέδου 5) θα χρειαστεί να εκκινήσετε το LiveFS από το CD και να δώσετε μερικές εντολές ώστε να αποκαταστήσετε ξανά τον... διεγραμμένο πυρήνα! (Πιο πολύ καφέ την επόμενη φορά, εντάξει;)

Όπως εξάλλου δεν υπάρχουν μαγικά βήματα, έτσι **δεν υπάρχουν και μαγικά αρχεία με μαγικές θέσεις**. Δεν μας ενδιαφέρει σε ποιο sector

βρίσκεται ο πυρήνας. Τα αρχεία έχουν ονόματα, permissions, flags, περιεχόμενα. Αν αυτά είναι OK, η φυσική τους θέση είναι αδιάφορη. Ουσιαστικά, το μόνο μαγικό κομμάτι είναι τα boot blocks (boot sector, loader). Αν λοιπόν ξέρουμε να εγκαταστήσουμε αυτά χειροκίνητα σε έναν νέο δίσκο κι αντιγράψουμε το υπόλοιπο λειτουργικό, όλα θα λειτουργήσουν μια χαρά!

Μην τρέξετε όμως να κάνετε cp τα αρχεία. Υπάρχουν πολύ καλύτερες εντολές, τη στιγμή μάλιστα που θέλουμε να διατηρήσουμε όλες τις λεπτομέρειες (τα metadata, αν θέλετε):

- Symbolic / hard links
- File ownerships, permissions
- Flags (sticky bit κ.λπ.)

Dump & Restore: Οι καλοί φίλοι του root, από τα παλιά!

Δεν φαντάζεστε από πόσο παλιά. Αλλά αν διαβάσετε το man page της dump, της εντολής που θα χρησιμοποιήσουμε για να πάρουμε ένα αντίγραφο του filesystem, θα βρείτε μέσα αναφορές σε ταινίες, tape changers κι άλλες εξωτικές συσκευές. (Όχι, δεν μπορείτε να πάρετε backup σε διάτρητες κάρτες). Βέβαια, στις μέρες μας το πιθανότερο είναι να πάρετε backup σε κάποια συνηθισμένη συσκευή: Ένα εξωτερικό USB flash drive ή δίσκο, έναν δεύτερο εσωτερικό δίσκο ή ακόμη και σε ένα σύστημα αρχείων που έχετε προσαρτήσει δικτυακά.

Dump σε εξωτερικό USB flash drive

Για λόγους απλότητας, θα θεωρήσουμε ότι έχετε ήδη διαμορφώσει το USB σας με FAT32. Είναι εκείνο το παλιό USB που μεταφέρετε από υπολογιστή σε υπολογιστή στη δουλειά σας κι έχει μέσα τουλάχιστον δέκα autorun.inf με τους αντίστοιχους ιούς, αλλά αυτό καθόλου δεν μας απασχολεί στο FreeBSD. Αρκεί να έχει ελεύθερο χώρο. Πώς θα κάνατε μια αντιγραφή του FreeBSD root filesystem;

Θεωρούμε ότι έχετε μια πρόσφατη έκδοση του λειτουργικού FreeBSD (9.0 ή 9.1-RELEASE) και χρησιμοποιείτε UFS με Soft Updates journaling. Προσέξτε: Αν και είναι δυνατόν να πάρετε αντίγραφο του filesystem ενώ το σύστημά σας είναι σε πλήρη και κανονική λειτουργία, αυτό δεν συνίσταται. Την ώρα που κάνετε το backup τα δεδομένα πιθανώς ν' αλλάζουν κυριολεκτικά κάτω από τα πόδια σας. Η εντολή dump διαθέτει ειδική παράμετρο (-L) ώστε να λαμβάνετε backup από live filesystems (χρησιμοποιεί τη δυνατότητα snapshoting του UFS) αλλά δεν μπορείτε να τη χρησιμοποιήσετε αν έχετε Soft Updates journaling.

Πριν ξεκινήσετε, καλό είναι να ξέρετε κάποιες λεπτομέρειες για το σύστημα σας, όπως, π.χ., ποιο partition είναι το root filesystem. Δείτε το /etc/fstab και θα βρείτε κάποια καταχώριση του τύπου

```
# Device          Mountpoint   FStype   Options   Dump    Pass#
/dev/ada0p1       /            ufs      rw        1       1
```

ή

```
/dev/gpt/gproot   /            ufs      rw        1       1
```

αν έχετε χρησιμοποιήσει labels στην εγκατάσταση (ειδικά αν έχετε ακολουθήσει το άρθρο μου στο http://bit.ly/nobsdinst). Και οι δύο μορφές μάς κάνουν.

Σας προτείνουμε λοιπόν να εκκινήσετε από το CD εγκατάστασης του FreeBSD και να επιλέξετε τη λειτουργία LiveCD. Κάντε login ως root (πιέστε απλώς [Enter] στο password) και είμαστε έτοιμοι.

Συνδέστε το flash drive. Το όνομα συσκευής θα το βρείτε εύκολα, δίνοντας

```
# ls /dev/da*
```

Εναλλακτικά, παρατηρήστε το αποτέλεσμα της εντολής

```
# dmesg | tail
```

Είναι πολύ πιθανόν να είναι, π.χ., το da0s1. Ας θεωρήσουμε επίσης ότι το root filesystem είναι το /dev/ada0p1. Πριν γράψουμε κάτι στο flash drive, θα χρειαστεί να το προσαρτήσουμε. Το περιβάλλον του LiveCD μάς δίνει έναν κατάλογο, τον /mnt, γι' αυτή τη δουλειά:

```
# mount -t msdosfs /dev/da0s1 /mnt
```

Είμαστε έτοιμοι για το dump:

```
# dump -0 -a -h0 -f /mnt/rootfs.dump /dev/ada0p1
```

Ας δούμε λίγο τις παραμέτρους.

-0: Level 0 dump. H dump μπορεί να εκτελέσει incremental backups. Το level 0 dump σημαίνει πλήρες backup. Ένα level 1 dump θα πάρει όλα τα αρχεία που έχουν αλλάξει σε σχέση με το προηγούμενο level 0 dump, ένα level 2 όσα έχουν αλλάξει από το level 1 και πάει λέγοντας. Εμείς, προφανώς, χρειαζόμαστε level 0 dump.

-a: Αυτόματη ανίχνευση του τέλους της ταινίας ή γενικά του μέσου στο οποίο γράφεται το dump.

-f: Γράφουμε το dump σε αρχείο κι εδώ δίνουμε το όνομα του.

/dev/ada0p1: Το (original) partition που κάνουμε dump. Προσέξτε ότι αν έχουμε πολλαπλά partitions θα πρέπει να τα κάνουμε dump χωριστά. Αν, για παράδειγμα, το /home μας είναι μια προσάρτηση από το /dev/ada0p2, αν το επιθυμούμε θα πρέπει να το κάνουμε χωριστά dump. Αν κάνουμε dump από το σύστημα μας την ώρα που λειτουργεί κανονικά, μπορούμε αντί για όνομα συσκευής να χρησιμοποίησουμε το σημείο προσάρτησης (π.χ., / ή /home). Το dump θα δει το /etc/fstab μας. Τώρα που είμαστε σε LiveCD, αυτό προφανώς δεν γίνεται.

-h0 (το καλύτερο για το τέλος): Τι γίνεται αν κάποια αρχεία δεν θέλουμε να τα πάρουμε στο backup μας; Ίσως δεν το ξέρετε, αλλά υπάρχει το 'nodump' flag, το οποίο μπορείτε να ορίσετε για αρχεία ή καταλόγους που δεν θέλετε να καταλήξουν στο dump σας. Για παράδειγμα, ίσως δεν θέλετε να πάρετε backup το /usr/ports. Τι να το κάνετε, άλλωστε; Αφού μπορείτε να το έχετε ξανά, με ένα portsnap fetch extract. Πληκτρολογήστε, λοιπόν, τα εξής:

```
# cd /usr
# chflags -R nodump ports/
```

Με την παράμετρο -o μπορείτε να δείτε, μέσω της ls, τα διάφορα flags όπως είναι το nodump:

```
# ls -lod /usr/ports
drwxr-xr-x  69 root  wheel  nodump 1536 Apr 28 19:51 /usr/ports/
```

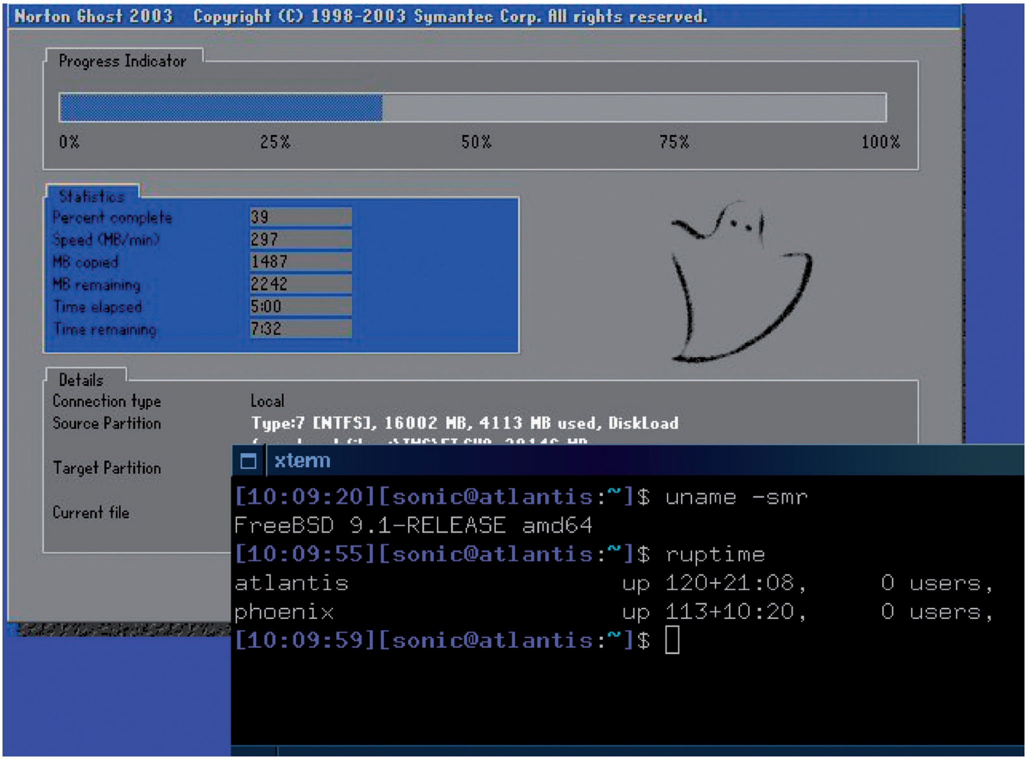
Η παράμετρος **-R** σημαίνει «κάνε αυτό που σου λέω στον κατάλογο και σ' όλα τα περιεχόμενα του» (recursively). Α, ναι, αυτό με τα flags θα πρέπει να το κάνετε με το σύστημα σας σε κανονική λειτουργία. Προλαβαίνετε να πιέσετε [CTRL+C], τώρα :) Υπάρχουν κι άλλοι υποψήφιοι κατάλογοι για nodump flag.

- Αν το /home σας δεν είναι σε χωριστό partition και θέλετε να αντιγράψετε *μόνο* την εγκατάσταση κι όχι τα δεδομένα σας.
- Κάποιοι κατάλογοι μέσα στο /var, π.χ. οι /var/db/freebsd-update και /var/db/cache/pkg.
- Κατάλογοι με προσωρινά δεδομένα (π.χ., /tmp, /var/tmp). Προσέξτε βέβαια ότι αν δεν τους βάλετε στο dump, θα πρέπει μετά να τους αναδημιουργήσετε με τα σωστά δικαιώματα και flags (chmod 1777 /tmp /var/tmp).

Τώρα, παράμετρος -h0 λέει στη dump: «αυτό το nodump που βλέπεις θέλω να το τιμήσεις ακόμη και για level 0 dump». Μπορούμε, όπως καταλαβαίνετε, να ορίσουμε από ποιο level και μετά η dump δίνει σημασία στο nodump flag. Κι αυτό γιατί σε ένα πλήρες backup (level 0) ίσως θέλουμε να συμπεριλάβουμε όλα τα αρχεία.

Πάνω: *Ο λόγος για τον οποίο είναι περήφανος ο Γιάννης.*

Κάτω: *Ο λόγος για τον οποίο εμείς δεν του δίνουμε καμιά απολύτως σημασία!*



Μέχρι να σας τα εξηγήσω όλα αυτά, πιστεύω ότι το dump σας τελείωσε. Αποπροσαρτήστε το flash drive κι επανεκκινήστε το σύστημα:

```
# umount /mnt
# shutdown -r now
```

Βγάλτε και το CD από τον οδηγό και πάμε στο νέο σας μηχάνημα, να κάνουμε restore την εγκατάσταση.

Ένα ευχάριστο απόγευμα με την restore

Η restore είναι πολύ εύκολη. Το ευχάριστο απόγευμα θα το περάσετε διαβάζοντας πώς γίνεται το partitioning. Χειροκίνητα, that is!

Παραδοσιακά, το FreeBSD χρησιμοποιούσε δύο εργαλεία: το fdisk και το bsdlabel. Το πρώτο έφτιαχνε τα slices (primary partitions, στα υπόλοιπα λειτουργικά) και το δεύτερο χώριζε κάθε slice σε μικρότερα κομμάτια (οι γνωστοί χώροι για τα /, /usr, /var, /tmp και το swap space). Στον κόσμο του FreeBSD αυτά τα κομμάτια ονομάζονται partitions και μπορείτε να τα σκέφτεστε σαν τα logical drives των Windows. Αν και θα μπορούσαμε να σας δείξουμε πώς χρησιμοποιούνται αυτές οι δύο εντολές, δεν θα το κάνουμε. Όλες οι σύγχρονες εκδόσεις του FreeBSD υποστηρίζουν το νέο GPT partitioning scheme, το οποίο υλοποιείται εξ' ολοκλήρου με μια και μοναδική εντολή: την gpart.

Με το GPT μπορούμε ν' απαλλαγούμε από τον περιορισμό των τεσσάρων μόνο slices. Έτσι, δεν χρειάζεται να κάνουμε και τη διαδικασία με την bsdlabel. Αν χρησιμοποιήσατε το bsdinstall για την τελευταία σας εγκατάσταση, έχετε ήδη GPT partitions.

Ας εκκινήσουμε πάλι με τη λειτουργία LiveCD αλλά στο νέο μας σύστημα. Θα βρούμε πρώτα το όνομα της συσκευής δίσκου (εδώ υποθέτουμε ότι είναι το ada0):

```
# ls /dev/ad*
/dev/ada0
```

Ας σιγουρευτούμε ότι ο δίσκος είναι άδειος (μην ανησυχήσετε αν πάρετε μήνυμα λάθους στην εντολή):

```
# gpart destroy -F ada0
ada0 destroyed
```

Δημιουργούμε gpt partition scheme (η εντολή gpart μπορεί να χρησιμοποιηθεί και για τη δημιουργία MBR partition scheme, αλλά προφανώς δεν μας ενδιαφέρει):

```
# gpart create -s gpt ada0
ada0 created
```

Είμαστε έτοιμοι να δημιουργήσουμε τα partitions μας. Θα δώσουμε 4GB για swap και το υπόλοιπο στο root partition. Δεν θα φτιάξουμε χωριστό /home (αλλά σας καλούμε να πειραματιστείτε με όλες τις δυνατότητες). Θα ξεκινήσουμε φτιάχνοντας το μόνο «μαγικό» partition τύπου freebsd-boot, που περιέχει τα boot blocks:

```
# gpart add -t freebsd-boot -l gpboot -s 128K ada0
ada0p1 added
# gpart bootcode -b /boot/pmbr -p /boot/gptboot -i 1 ada0
bootcode written to ada0
```

Συνεχίζουμε με το swap:

```
# gpart add -t freebsd-swap -l gpswap -b 1M -s 4G ada0
ada0p2 added
```

Τελειώνουμε με το root partition, στο οποίο θα δώσουμε όλον τον υπόλοιπο χώρο. Σ' αυτή την περίπτωση, απλώς παραλείπουμε εντελώς την παράμετρο -s:

```
# gpart add -t freebsd-ufs -l gproot ada0
ada0p3 added
```

Δημιουργούμε το σύστημα αρχείων για το root, με Soft Updates journaling:

```
# newfs -j /dev/gpt/gproot
...
newfs: soft updates journaling set
```

Θα προσαρτήσουμε το νέο μας partition στο /media και το USB drive (είναι ώρα να το συνδέσετε) στο /mnt. όπως και πριν:

```
# mount -t msdosfs /dev/da0s1 /mnt
# mount /dev/gpt/gproot /media
```

Και είμαστε έτοιμοι να ξεκινήσουμε το restore:

```
# cd  /media
# env TMPDIR=/media restore -rvf /mnt/rootfs.dump
```

Οι παράμετροι είναι σχεδόν προφανείς:

-r: Restore file system :)

-v: Verbose output. Εμφάνιση των αρχείων κατά τη διαδικασία αποκατάστασης.

-f: Το όνομα του αρχείου που περιέχει το dump.

Η αποκατάσταση γίνεται στον τρέχοντα κατάλογο, για το λόγο αυτό υπάρχει και η εντολή cd /media. Για τη λειτουργία του restore χρειάζεται μια περιοχή tmp. Για να είμαστε σίγουροι, δίνουμε ως TMPDIR τον κατάλογο που κάνουμε το restore. Με το τέλος της διαδικασίας, χρησιμοποιήστε έναν editor όπως ο vi ή ο ee (και οι δύο είναι στο base system και υπάρχουν στο LiveCD) για να κάνετε μερικές μικρές τροποποιήσεις.

- Να αλλάξετε την γραμμή ifconfig στο /media/etc/rc.conf, ώστε η κάρτα δικτύου να ταιριάζει με αυτή του νέου μηχανήματος. Είναι επίσης μια καλή ευκαιρία να απενεργοποιήσετε προσωρινά τυχόν αυτόματο login σε περιβάλλον γραφικών, ώστε να φτιάξετε νέο xorg.conf μετά την πρώτη εκκίνηση. Θα βρείτε σίγουρα και άλλα πράγματα να αλλάξετε (π.χ., το hostname).
- Να διορθώσετε το fstab:

```
# ee /media/etc/fstab

# Device          Mountpoint     FStype    Options    Dump    Pass#
/dev/gpt/gpswap   none           swap      sw         0       0
/dev/gpt/gproot   /              ufs       rw         1       1
```

Για να δείτε τι καλοί που είμαστε, σας έχουμε ετοιμάσει ήδη ένα dump για να παίξετε άφοβα, είτε σε κανονικό μηχάνημα είτε σε VMware/VirtualBox. Κατεβάστε το άφοβα από εδώ:

http://www.freebsdworld.gr/files/rootfs.dump.xz

Το αρχείο είναι συμπιεσμένο αλλά δεν χρειάζεται να το αποσυμπιέσετε για να κάνετε restore. Απλώς, η εντολή σας θα είναι η παρακάτω:

```
# cd /media
# env TMPDIR=/media xz -dc /mnt/rootfs.dump.xz | restore -rvf -
```

Εδώ, το xz αποσυμπιέζει στο standard output, το οποίο γίνεται pipe στο standard in της εντολής restore. Το «-» μετά το f δηλώνει ότι η restore δεν θα διαβάσει κάποιο πραγματικό αρχείο αλλά το standard input. Για επιπλέον ρυθμίσεις και πληροφορίες σχετικά με το έτοιμο αυτό image, δείτε στη σχετική εικόνα.

Τελειώνοντας με τις αλλαγές, αποπροσαρτήστε τα συστήματα αρχείων κι επανεκκινήστε:

```
# cd /root
# umount /media
# umount /mnt
# shutdown -r now
```

Μα, εγώ θέλω το νέο μου σύστημα Full ZFS!

Αχ αυτή η νεολαία, τα θέλει όλα και τα θέλει τώρα. Ε, ναι λοιπόν, γίνεται να μεταφέρετε το σύστημα σας σε ZFS. Απλώς, σε μια τέτοια περίπτωση τα πράγματα θα είναι κάπως πιο πολύπλοκα. Τα παρακάτω βασίζονται γενικά σε πληροφορίες που θα βρείτε στο FreeBSD wiki:

https://wiki.freebsd.org/RootOnZFS/GPTZFSBoot/9.0-RELEASE

Για να μη σας τρομάξω με το πλήθος των εντολών που πρέπει να γράψετε, αποφάσισα να σας δώσω δύο μικρά scripts που θα κάνουν όλη σχεδόν τη δουλειά για σας. Καλό θα είναι φυσικά να τα μελετήσετε και να τα μετατρέψετε για τις δικές σας ανάγκες, σαν καλοί διαχειριστές που είστε! Κατεβάστε τα από το

http://www.freebsdworld.gr/files/zfs-scripts.zip

Αποσυμπιέστε το zip και θα βρείτε δύο αρχεία: Το **prepare-zfs** και το **after-restore**. Αντιγράψτε τα κι αυτά στο USB flash, μαζί με το dump. Ξεκινήστε με το LiveCD όπως πριν, συνδέστε το USB και προσαρτήστε το:

```
# mount -t /dev/da0s1  /mnt
```

Εκτελέστε τώρα το script zfs-prepare:

```
# cd /mnt
# sh zfs-prepare ada0 4G
```

Το script δέχεται δύο παραμέτρους: Το όνομα της συσκευής δίσκου και το μέγεθος του swap partition που θα δημιουργηθεί. Θα φτιάξει τόσο τα partitions όσο και το σύστημα αρχείων ZFS, έτοιμο για να εκτελέσετε το restore.

Κάντε τώρα το restore όπως προηγουμένως. Το σύστημα αρχείων ZFS

έχει ήδη προσαρτηθεί, προσωρινά, στο /media. Με το τέλος του restore θα χρειαστείτε ακόμα μερικά βήματα. Προσθέστε τη γραμμή

```
zfs_enable="YES"
```

στο αρχείο /media/etc/rc.conf. Προσθέστε επίσης τη γραμμή

```
zfs_load="YES"
```

στο αρχείο /media/boot/loader.conf. Το αρχείο /media/etc/fstab χρειάζεται να έχει πλέον μόνο την παρακάτω γραμμή (ο πυρήνας ενημερώνεται διαφορετικά για τη θέση του root filesystem):

```
# Device          Mountpoint   FStype   Options   Dump   Pass#
/dev/gpt/gpswap   none         swap     sw        0      0
```

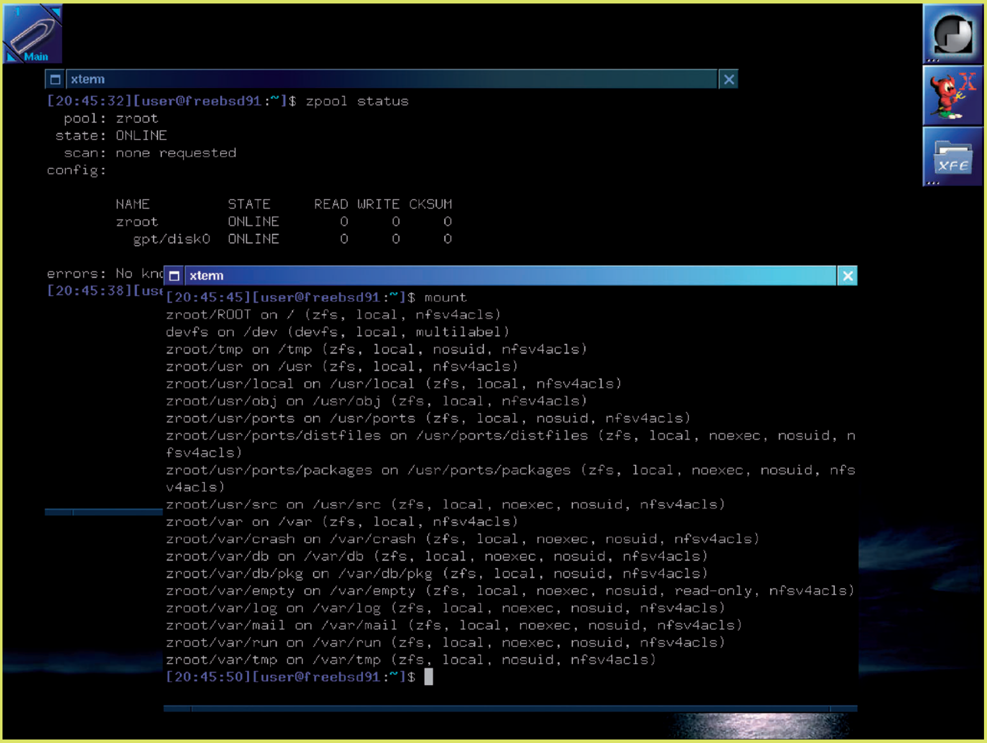
Έχοντας τελειώσει με τα παραπάνω, η οριστικοποίηση των ρυθμίσεων γίνεται με την εκτέλεση του δεύτερου script:

```
# cd /mnt
# sh after-restore
```

Επανεκκινήστε το σύστημα κρατώντας την αναπνοή σας: Αν τα κάνατε όλα σωστά, θα έχετε ένα πλήρες FreeBSD/ZFS σύστημα. Προφανώς, μπορείτε να κάνετε τη δοκιμή σας με το έτοιμο image που σας δίνουμε, σε VMware ή σε VirtualBox. Φροντίστε μόνο να δώσετε αρκετή μνήμη στο VM —προτείνουμε τουλάχιστον 1GB—, καθώς το ZFS είναι μνημοβόρο!

Ελπίζω να περάσετε πολλά ευχάριστα απογεύματα, γεμίζοντας τον κόσμο FreeBSD μηχανήματα!





Πάρτε από το http://www.freebsdworld.gr/files/rootfs.dump.xz το έτοιμο dump που σας δίνουμε, για να δοκιμάσετε όλους τους πιθανούς τρόπους restore, χωρίς να προκαλέσετε ΔΑΛΕΚ επιπέδου 5 στο βασικό σας μηχάνημα! Προφανώς, μπορείτε να το χρησιμοποιήσετε τόσο σε VMware/VirtualBox όσο και σε κάποιο έξτρα μηχάνημα που έχετε. Και φυσικά, αν θέλετε μπορείτε να χρησιμοποιήσετε τις οδηγίες μας για full ZFS. Το έτοιμο dump περιέχει μόνο βασικές εφαρμογές και το περιβάλλον γραφικών του Windowmaker. Ύπάρχουν δύο χρήστες, ο root και ο user, με κοινό password το υπερασφαλές 1234 που μάλλον θα θελήσετε ν' αλλάξετε. Το ωραίο όμως είναι ότι το σύστημα από το οποίο προέρχεται το dump έχει δημιουργηθεί με έτοιμα πακέτα τύπου PkgNG (http://deltahacker.gr/?p=7358). Αν ρίξετε μια ματιά στο /usr/local/etc/pkg.conf θα δείτε ότι μπορείτε να εγκαταστήσετε κι άλλα, από το (unofficial) repository που έχουμε ρυθμίσει.

Ενέργειες για εσάς μετά την πρώτη εκκίνηση:

- Συνδεθείτε ως root. Αν κάνατε restore σε VMware ή σε κανονικό, φυσικό μηχάνημα, από το /etc/rc.conf αφαιρέστε τις γραμμές που αφορούν στο VirtualBox.
- Αλλάξτε τον κωδικό του root.
- Δημιουργήστε ένα νέο xorg.conf και μεταφέρετέ το στην τελική του θέση (περιέχονται drivers για VMware και VirtualBox):

```
# X -configure
# mv /root/xorg.conf.new /etc/X11/xorg.conf
```

- Συνδεθείτε ως user, αλλάξτε τον κωδικό και δώστε startx.
- Ασχοληθείτε και χαρείτε για ώρες!

Συνταγές... παραμόρφωσης!

Arduino Sound Master

part 6/κάπου τόσο

Skill: Intermediate

Tags: SID, ring modulation, vibrato, tremolo, flange, arpeggio

Το SID διέθετε εξειδικευμένο κύκλωμα για την υλοποίηση ενός μόνο εφέ. Ωστόσο, αποτέλεσε τη βάση για την ανάπτυξη πολλών ακόμα, σε προγραμματιστικό επίπεδο. Όλα αυτά τα εφέ, κάποια λιγότερο και κάποια περισσότερο, ευθύνονται για τον χαρακτηριστικό ήχο των παιχνιδιών της εποχής! Τι θα λέγατε να τα γνωρίσουμε και να παίξουμε μαζί τους;



by Spir@l Evolut10n

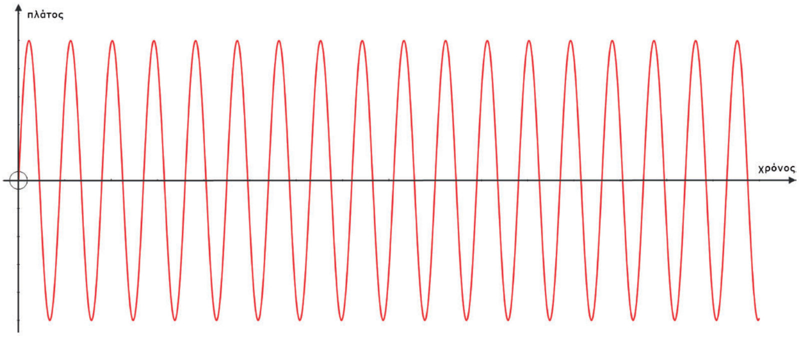
Στο προηγούμενο άρθρο της σειράς, στο deltaHacker 019, γνωρίσαμε το τσιπάκι ήχου των Commodore 64, το περίφημο SID. Ξεκινήσαμε με τα βασικά χαρακτηριστικά του, μελετήσαμε τα τέσσερα είδη κυμάτων που μπορούσε να συνθέσει και κατασκευάσαμε ένα μικρό synthesizer που παρήγαγε παρόμοιους ήχους. Ωστόσο, η μουσικολογική-προγραμματιστική περιπλάνηση στην εποχή των πρώτων ηλεκτρονικών παιχνιδιών δεν θα μπορούσε να ολοκληρωθεί μ' αυτήν την επιφανειακή ενασχόληση με το SID. Το τσιπάκι ενσωμάτωνε κύκλωμα για την υλοποίηση ενός μόνο ηχητικού εφέ (ring modulation), ωστόσο επέτρεπε στους προγραμματιστές να αναπτύσσουν πολλά ακόμα. Επομένως, αν θέλουμε να λέμε ότι γνωρίζουμε τα μυστικά του ήχου εκείνης της εποχής, πρέπει να κατανοήσουμε τη φύση αυτών των ηχητικών εφέ και να μάθουμε πώς υλοποιούνταν.

Ανάγκη που έγινε έμπνευση

Πριν αναφερθούμε στα διασημότερα ηχητικά εφέ και στον τρόπο παραγωγής του καθενός, πρέπει να πούμε δύο λόγια για την ανάγκη που ώθησε τους προγραμματιστές να τα δημιουργήσουν. Τα τσιπάκια ήχου της εποχής προσέφεραν ελάχιστα μέσα στους μουσικούς. Ακόμα και το SID, ο βασιλιάς των τσιπ ήχου, μπορούσε να δημιουργήσει μόλις τέσσερα διαφορετικά είδη κυμάτων. Σκεπτόμενοι αφηρημένα, θα μπορούσαμε να θεωρήσουμε καθένα από αυτά τα κύματα σαν ένα διαφορετικό όργανο. Επομένως, θα μπορούσαμε να πούμε ότι το SID διέθετε τέσσερα διαφορετικά όργανα. Όπως αντιλαμβάνεστε, τέσσερα όργανα με τόσο φτωχικό αλλά και παραπλήσιο ήχο, δεν αφήνουν πολλά περιθώρια για τη σύνθεση μουσικών κομματιών. Κάπως έτσι, εφόσον το hardware δεν παρείχε έτοιμες λύσεις, το βάρος μετατέθηκε στους προγραμματιστές: Ήταν δικό τους καθήκον να αυξήσουν τα διαθέσιμα όργανα, ώστε οι μουσικοί να αποκτήσουν μεγαλύτερη ελευθερία έκφρασης. Ουσιαστικά, οι προγραμματιστές έπρεπε να δημιουργήσουν νέους ήχους, αλλοιώνοντας εκείνους που προσέφερε ήδη το hardware. Όμως, πώς μπορεί να αλλοιωθεί ο ήχος ενός κύματος; Η απάντηση είναι απλή: Μεταβάλλοντας ταχύτατα ένα ή περισσότερα χαρακτηριστικά του. Για παράδειγμα, για να δημιουργήσουμε νέους ήχους από ένα τετραγωνικό κύμα (square ή pulse wave), μπορούμε να μεταβάλλουμε με μεγάλη ταχύτητα το Duty Cycle, τη συχνότητα, το πλάτος ή και τη φάση του.

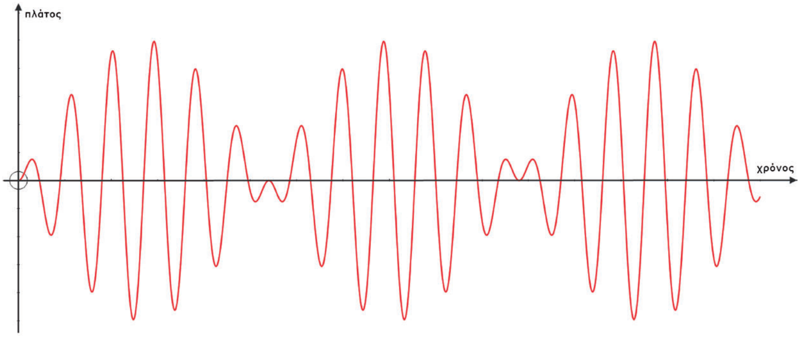
Διάσημα ηχητικά εφέ

Η λύση που περιγράψαμε για τη δημιουργία νέων ήχων μοιάζει απλοϊκή. Δικαίως, κάποιος θα μπορούσε να σκεφτεί ότι είναι αδύνατο να δώσει πολλά διαφορετικά αποτελέσματα. Ως συνήθως, βέβαια, αυτός ο «κάποιος» κάνει λάθος. Βλέπετε, η μεταβολή ενός χαρακτηριστικού ενός κύματος μπορεί να είναι γραμμική, λογαριθμική, εκθετική ή αρμονική, ενώ θα μπορούσε να περιγράφεται κι από έναν συνδυασμό των παραπάνω! Ας εξετάσουμε όμως μια απλή περίπτωση: Έστω ότι έχουμε ένα τριγωνικό κύμα, του οποίου η συχνότητα ξεκινά από μια τιμή και καταλήγει σε κάποια άλλη, ενώ μεταβάλλεται γραμμικά. Η ποιότητα (το άκουσμα) του ήχου που θα προκύψει, δεν εξαρτάται μόνο από τα όρια αυτής της γραμμικής μεταβολής αλλά κι από την ταχύτητά της. Επομένως, πέρα από τη «μορφή» που παρουσιάζει η μεταβολή, σημασία έχει και η ταχύτητά με την οποία εξελίσσεται. Τελικά, η λύση που είχαν σκαρφιστεί οι προγραμματιστές για τη δημιουργία νέων ήχων άνοιγε πολλούς ορίζοντες. Για του λόγου το αληθές, θα δούμε μερικά διάσημα ηχητικά εφέ.



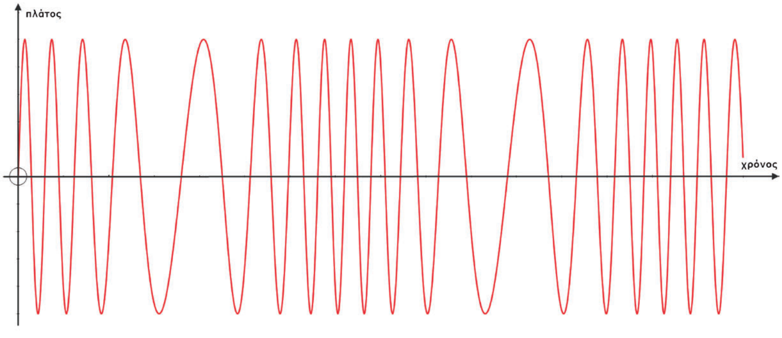
Εικόνα 1
Στο πρώτο άρθρο της σειράς, στο deltaHacker 015, παιδευτήκαμε για να συνθέσουμε ένα απλό ημιτονοειδές κύμα. Τώρα θα παιδευτούμε ακόμα περισσότερο, αλλά ταυτόχρονα θα παιδέψουμε και το ταλαίπωρο κύμα. Δείτε στις επόμενες εικόνες τι θα του προκαλέσουν τα διάφορα εφέ.

- **Tremolo:** Αυτό το εφέ προβλέπει την περιοδική μεταβολή του πλάτους και άρα της έντασης ενός ηχητικού κύματος. Συνήθως η εν λόγω μεταβολή είναι αρμονική (ημιτονοειδής) και το αποτέλεσμά της θυμίζει αυτό που στη φυσική ονομάζουμε διακρότημα (εικόνα 2).



Εικόνα 2
Στο tremolo το πλάτος του κύματος μεταβάλλεται περιοδικά, με τον τρόπο που καθορίζει η μορφή και η συχνότητα ενός άλλους κύματος. Στην εικόνα, η μεταβολή του πλάτους του κύματος είναι αρμονική (ημιτονοειδής) κι έχει περίπου έξι φορές μικρότερη συχνότητα σε σύγκριση με εκείνη του κύματος...

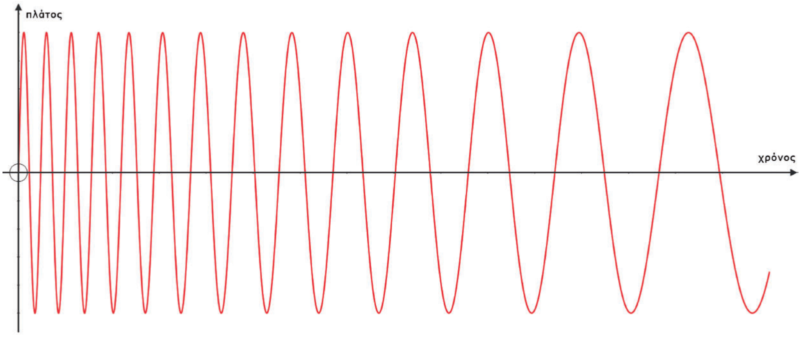
- **Vibrato:** Αυτή τη φορά δεν μεταβάλλεται το πλάτος αλλά η συχνότητα ενός κύματος (εικόνα 3).



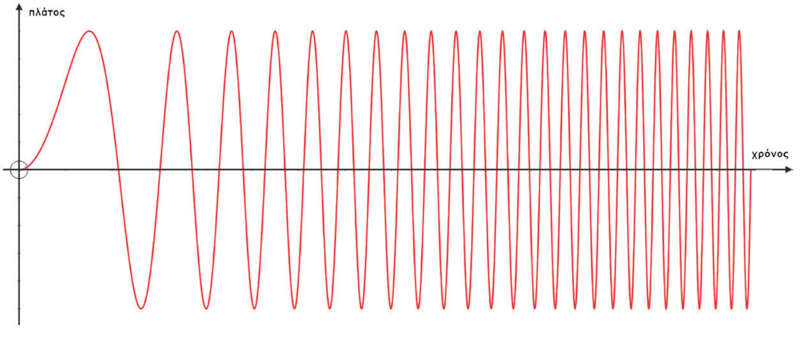
Εικόνα 3
Το vibrato προβλέπει την περιοδική μεταβολή της συχνότητας ενός κύματος. Στην εικόνα, η συχνότητα του κύματος ξεκινά από κάποια τιμή, στη συνέχεια μειώνεται και μετά αυξάνεται εκ νέου. Αυτός ο «κύκλος» επαναλαμβάνεται αδιάκοπα...

- **Glide & Portamento:** Αυτά τα εφέ προβλέπουν τη γραμμική μεταβολή της συχνότητας μιας νότας, μεταξύ δύο τιμών. Το glide περιγράφει τη μεταβολή της συχνότητας από την τιμή της εκάστοτε νότας ως μια άλλη, αυθαίρετα ορισμένη τιμή. Αντίθετα, το portamento περιγράφει τη μεταβολή της συχνότητας από μια αυθαίρετα ορισμένη τιμή ως εκείνη της εκάστοτε νότας. Έτσι, το glide εφαρμόζεται στο «τέλος» μιας νότας, ενώ το portamento στην

«αρχή» (εικόνες 4 και 5). Αυτά τα δύο εφέ, ανάλογα με το αν η συχνότητα μεταβάλλεται προς μια χαμηλότερη ή υψηλότερη τιμή, συναντώνται κι ως pitch-bend down ή pitch-bend up.



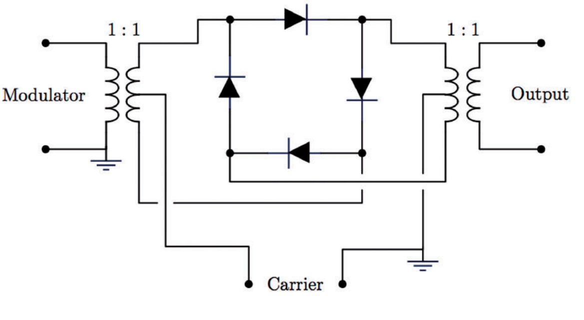
Εικόνα 4
Προς το «τέλος» της νότας, λίγο πριν πάψει η αναπαραγωγή της, η συχνότητα αρχίζει να μειώνεται. Αυτό το εφέ ονομάζεται glide και αποτελεί ειδική περίπτωση του pitch-bend down.

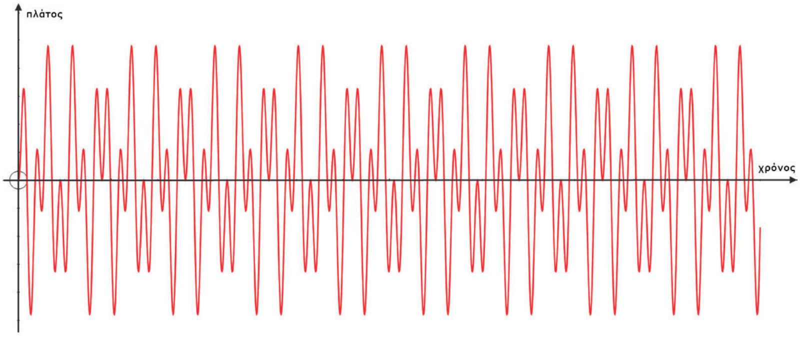


Εικόνα 5
Στην «αρχή» της νότας, με το που αρχίζει η αναπαραγωγή της η συχνότητα ξεκινά από μια χαμηλή τιμή κι αυξάνει ως την τιμή της εκάστοτε νότας. Αυτό το εφέ ονομάζεται portamento κι αποτελεί ειδική περίπτωση του pitch-bend up.

- **Ring modulation:** Εδώ τα πράγματα περιπλέκονται λίγο, αφού η υλοποίηση αυτής της τεχνικής από τσιπάκι σε τσιπάκι διαφέρει. Σε γενικές γραμμές, το ring modulation προβλέπει τη μίξη δύο κυμάτων. Από αυτά, εκείνο που καθορίζει το εφέ έχει συνήθως πολύ μεγαλύτερη συχνότητα από εκείνο στο οποίο επιβάλλουμε το εφέ. Το όνομα προέκυψε από τη μορφή του αντίστοιχου αναλογικού κυκλώματος (εικόνα 6), το οποίο στη θεωρητική του εκδοχή μοιάζει να περιλαμβάνει ένα δαχτυλίδι από διόδους. Στην περίπτωση του SID, το ring modulation μπορούσε να εφαρμοστεί μόνο στα τριγωνικά κύματα. Η υλοποίηση στο SID προέβλεπε την αντιστροφή του τριγωνικού κύματος, όταν το κύμα που διαμόρφωνε το εφέ βρισκόταν στη μία από τις δύο ημιπεριόδους του. Με αυτόν τον τρόπο, το τριγωνικό κύμα έμοιαζε να «σπάει» κάθε τόσο και η φάση (phase) πραγματοποιούσε απότομες μετατοπίσεις. Αυτή η περιγραφή όμως είναι αρκετά θεωρητική και κάτι μάς λέει ότι δεν βοηθά ιδιαίτερα. Κρατείστε ότι το ring modulation παράγει ήχους που μοιάζουν να προέρχονται από καμπανάκια, ενώ ειδικά η υλοποίησή του στο SID παράγει ως επί το πλείστον μεταλλικούς ήχους.

Εικόνα 6
Το αναλογικό κύκλωμα που υλοποιεί το ring modulation, στη θεωρητική του μορφή. Το όνομα του εφέ προέκυψε από τη διάταξη ων τεσσάρων διόδων στο κέντρο του κυκλώματος.





Εικόνα 7
Το ring modulation προβλέπει τη μίξη δύο κυμάτων. Με αυτή τη διαδικασία προκύπτει ένα νέο κύμα, το οποίο περιέχει το άθροισμα και τη διαφορά των συχνοτήτων των δύο αρχικών κυμάτων. Χαμός!

Παιχνίδια με το χρόνο

Τα παραπάνω εφέ ήταν πολύ διαδεδομένα και σε παραλλαγές ή συνδυασμούς αυτών είχαν βασιστεί πολλά ακόμα. Ωστόσο, υπάρχουν και δύο πολύ γνωστά εφέ, τα οποία δεν σχετίζονται καθόλου με τα προηγούμενα. Αναφερόμαστε στα arpeggio και flange.

- **Arpeggio:** Υπό φυσιολογικές συνθήκες, οι νότες μιας συγχορδίας αναπαράγονται ταυτόχρονα. Κατά συνέπεια, για να αναπαραχθεί μια συγχορδία με τρεις νότες, το εκάστοτε πρόγραμμα θα έπρεπε θα δεσμεύσει ισάριθμα κανάλια ήχου. Τις περισσότερες φορές, όμως, κάτι τέτοιο ήταν απλά αδύνατο[1]. Έτσι, οι προγραμματιστές κατέφυγαν σε μια τεχνική που εφαρμόζεται στα (φυσικά) μουσικά όργανα κι ονομάζεται άρπισμα. Σύμφωνα μ' αυτή, οι νότες μιας συγχορδίας δεν παίζονται ταυτόχρονα αλλά διαδοχικά. Το άρπισμα απαιτεί σχετικά μεγάλη ταχύτητα από τον μουσικό αλλά, όπως καταλαβαίνετε, αυτό δεν αποτελεί πρόβλημα για έναν υπολογιστή. Τελικά, η τεχνική του arpeggio επέτρεπε στους προγραμματιστές να παίζουν συγχορδίες δεσμεύοντας μόνο ένα κανάλι, στο οποίο αναπαράγονταν διαδοχικά και με μεγάλη ταχύτητα οι αντίστοιχες νότες. Το arpeggio εμφανιζόταν πολύ συχνά στη μουσική των παλιών παιχνιδιών και, μαζί με το ring modulation, αποτελούν τα πιο χαρακτηριστικά γνωρίσματα αυτής της μουσικής!
- **Flange:** Αυτό το εφέ μπορούσε να ζωντανέψει μια μελωδία, αφού άλλοτε προσέδιδε στη μελωδία ηχώ (με μεταβαλλόμενο μάλιστα βάθος) κι άλλοτε μια κάποια μεταλλική υφή. Η επίδραση αυτού του εφέ σε μια μελωδία είναι περίπλοκη και μεταβάλλεται με το χρόνο. Το πώς επιδρά το flange στο ηχητικό φάσμα μιας μελωδίας περιγράφεται δύσκολα. Ακόμη και μετά την περιγραφή, δεν βοηθούμαστε ιδιαίτερα στην κατανόησή του[2]. Το μόνο σίγουρο είναι ότι αν έχετε παίξει παιχνίδια της δεκαετίας του '80 έχετε ακούσει το flange. Για την υλοποίησή του απαιτείται η δέσμευση δύο καναλιών, τα οποία αναπαράγουν την ίδια μελωδία. Ωστόσο, το ένα από τα δύο παρουσιάζει μια μικρή καθυστέρηση σε σχέση με το άλλο, η οποία μάλιστα δεν είναι σταθερή.

1. Είτε γιατί το hardware διέθετε λιγότερα κανάλια(!) είτε γιατί τα περισσότερα ήταν ήδη «κατειλημμένα» από άλλες μελωδίες και τα ελεύθερα δεν επαρκούσαν.
2. Το φάσμα συχνοτήτων του σήματος εμφανίζει αρκετές νέες κορυφές, οι οποίες μάλιστα μετατοπίζονται. Σας το είπαμε! Ούτε κι αυτή η περιγραφή βοηθά ;)

Στην πράξη

Μετά από τόση θεωρία, για να βεβαιωθούμε ότι την αφομοιώσαμε και κυρίως για να διασκεδάσουμε, θα εξετάσουμε ένα πρόγραμμα που υλοποιεί μερικά εφέ. Πριν ξεκινήσουμε την παρουσίαση του κώδικα, σημειώστε ότι μπορείτε να τον κατεβάσετε από το http://bit.ly/dh020sideffects. Σε αυτή την έκδοση του προγράμματος, το μικρό μας synthesizer αναπαράγει μόνο τριγωνικά κύματα. Ωστόσο, με τη βοήθεια ενός ποτενσιόμετρου μπορούμε να επιλέξουμε την εφαρμογή ενός από τα ακόλουθα τρία εφέ: Ring modulation, vibrato και tremolo. Το άλλο ποτενσιόμετρο έχει διπλό ρόλο: Όταν έχουμε επιλέξει το ring modulation, ρυθμίζει τη συχνότητα του κύματος που αναμιγνύεται με την εκάστοτε νότα. Όταν όμως έχουμε επιλέξει το vibrato, το εν λόγω ποτενσιόμετρο ρυθμίζει το πλάτος της ταλάντωσης, που πραγματοποιεί η συχνότητα της εκάστοτε νότας. Μπερδευτήκατε; Όπως και να ‘χει, θα σας προτείναμε να φτιάξετε το γνωστό κύκλωμα (εικόνα 8), να φορτώσετε το πρόγραμμά μας στο Arduino και να αρχίσετε να παίζετε! Όταν όμως βαρεθείτε (ή δεχτείτε παρατήρηση για τους παράξενους ήχους που βγαίνουν από το δωμάτιό σας), καλό θα ήταν να επιστρέψετε στον κώδικα και στη μελέτη του.

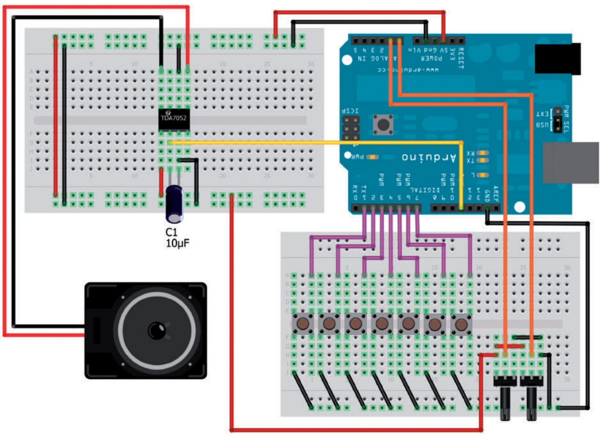
Η νέα εκδοχή του προγράμματος είναι βασισμένη σε μεγάλο βαθμό στις προηγούμενες. Οι κυριότερες αλλαγές εντοπίζονται στη ρουτίνα υπολογισμού των νέων δειγμάτων. Εκεί θα διακρίνετε αμέσως μια δομή switch, η οποία περιλαμβάνει τέσσερεις ενότητες (cases). Οι τρεις από αυτές υλοποιούν τα τρία εφέ του προγράμματός μας, ενώ η τέταρτη αφήνει το τριγωνικό κύμα αναλλοίωτο. Επιπρόσθετα, πάντα μέσα στην ίδια ρουτίνα, ο κώδικας διαβάζει τα δύο ποτενσιόμετρα κι ενημερώνει κατάλληλα τις μεταβλητές effect και depth. Η πρώτη καθορίζει το εφέ που εφαρμόζεται στο τελικό κύμα, ενώ η δεύτερη ρυθμίζει την ένταση του εφέ. Στη συνέχεια θα εξετάσουμε τα νέα και πιο ενδιαφέροντα τμήματα του κώδικα: αυτά στα οποία υλοποιούνται τα εφέ.

```
case 0: // ----- tremolo
if (sample_cnt == 45) {
steps++;
steps &= 31;
if (steps < 16) {if(volume > 8) {volume -= 8;}}
else {if(volume < 248) {volume += 8;}}
}
break;
```

Όπως ξεκαθαρίζει και το σχόλιο, εδώ υλοποιείται το tremolo. Ο κώδικας ξεκινά με έναν έλεγχο του μετρητή δειγμάτων (sample_cnt). Η τιμή (45) με την οποία συγκρίνεται η σχετική μεταβλητή, επιλέχθηκε *σχεδόν* τυχαία. Το ζητούμενο με αυτόν τον έλεγχο είναι να επιβραδύνουμε την εφαρμογή του εφέ, ώστε το πλάτος του τριγωνικού κύματος να αλλάζει μόνο μία φορά ανά 256 δείγματα. Αυτή η καθυστέρηση είναι αναγκαία, ώστε το tremolo να εξελίσσεται σχετικά αργά και να γίνεται εύκολα αντιληπτό! Το τέχνασμα με τον έλεγχο του μετρητή δειγμάτων χρησιμοποιείται και για την καθυστέρηση στην ανάγνωση των δύο ποτενσιόμετρων[3]. Αν ρίξετε μια ματιά σ' αυτή την περιοχή του κώδικα, θα διαπιστώσετε ότι τα ποτενσιόμετρα ελέγχονται όταν ο μετρητής δειγμάτων έχει τις τιμές 15 και 30. Στην περίπτωση των εφέ επιλέξαμε για τον μετρητή μια διαφορε-

3. Η συχνότερη ανάγνωσής τους θα καθυστερούσε την εκτέλεση της ρουτίνας και επί της ουσίας δεν θα προσέφερε τίποτα.

τική τιμή (45). Αυτό το κάναμε για να εξασφαλίσουμε ότι η εφαρμογή του εφέ δεν θα συμπίπτει χρονικά με την ανάγνωση των ποτενσιόμετρων και η ρουτίνα υπολογισμού των δειγμάτων δεν θα παρουσιάζει καθυστερήσεις. Κατά τα άλλα, σε αυτή την περιοχή του κώδικα χρησιμοποιείται ένας μετρητής (steps), ο οποίος μετρά συνεχώς από το 0 ως το 31. Στα μισά «βήματα» αυτού του μετρητή η μεταβλητή volume μειώνεται, ενώ στα άλλα μισά αυξάνεται. Όμως, με τη μεταβλητή volume πολλαπλασιάζονται όλα τα δείγματα του παραγόμενου κύματος. Ως εκ τούτου, η ένταση του τελικού κύματος αυξάνεται και μειώνεται περιοδικά.



Εικόνα 8
Το κύκλωμα για το οποίο προορίζεται το πρόγραμμά μας. Ουσιαστικά, είναι το ίδιο κύκλωμα στο οποίο βασιστήκαμε και στα προηγούμενα δύο άρθρα της σειράς. Λογικά, θα πρέπει να το έχετε ήδη έτοιμο ;)

```
case 1: // ----- vibrato
if (sample_cnt == 45) {
steps++;
steps &= 31;
if (steps < 8) {phase_delta -= phase_delta_delta;}
if ((steps > 7) && (steps < 24)) {phase_delta += phase_delta_
delta;}
if (steps > 23) {phase_delta -= phase_delta_delta;}
}
break;
```

Εδώ υλοποιείται το vibrato. Το συγκεκριμένο απόσπασμα παρουσιάζει την ίδια γενική διάρθρωση με το προηγούμενο: Αφενός, ο κώδικας εκτελείται μόνο ανά 256 δείγματα και η εξέλιξη του εφέ έχει επιβραδυνθεί τεχνητά. Αφετέρου, αξιοποιείται και πάλι ο μετρητής steps, για την περιοδική μεταβολή ενός μεγέθους. Το μέγεθος που αλλάζει αυτή τη φορά είναι η τιμή της μεταβλητής phase_delta, η οποία εκτελεί μια ταλάντωση γύρω από την αρχική της τιμή. Ως εκ τούτου, μια αντίστοιχη ταλάντωση πραγματοποιεί και η συχνότητα του τελικού κύματος! Παρατηρείστε ότι το phase_delta μεταβάλλεται κάθε φορά κατά ένα μέγεθος που ονομάζεται phase_delta_delta[4]. Αυτό το μέγεθος προκύπτει σαν κλάσμα του phase_delta και υπολογίζεται σε συνάρτηση με το επιλεγμένο depth. Έτσι, το ποτενσιόμετρο που ρυθμίζει τη μεταβλητή depth (την «ένταση» του εφέ), ρυθμίζει και το εύρος της διακύμανσης της συχνότητας. Σημειώστε ότι ο υπολογισμός του phase_delta_delta γίνεται κατά τέτοιον τρόπο, ώστε η τιμή της phase_delta να βρίσκεται πάντα εντός των αποδεκτών ορίων. Βλέπετε, υπό συγκεκριμένες συνθήκες[5] η μεταβλητή phase_delta θα μπορούσε να κάνει το λεγόμενο underflow[6] και η τιμή της θα μεταπηδούσε κοντά στο άνω όριο για τις μεταβλητές integer. Η απότομη μεταβολή του phase_delta θα είχε σαν συνέπεια μια αντίστοιχα απότομη μεταβολή στη συχνότητα του κύματος. Αυτή η συμπεριφορά, εκτός του

4. Είχαμε κέφια όταν γράφαμε τον κώδικα.
5. Για μεγάλες τιμές του phase_delta_delta και για μικρές τιμές του phase_delta.
6. Πρόκειται για το αντίθετο του overflow και, όχι, δεν σας κάνουμε πλάκα.

ότι δεν προβλέπεται από το vibrato παράγει θόρυβο και γενικότερα είναι ανεπιθύμητη.

```
case 2: // ----- ring modulation
if (phase.bytes[1] & (1<<depth)) {sample = ~sample;}
break;
```

Εδώ υλοποιείται το ring modulation και μάλιστα με τον ίδιο τρόπο που υλοποιείται στο SID: Κάθε φορά που το δεύτερο κύμα (αυτό που καθορίζει το εφέ) βρίσκεται σε μία από τις ημιπεριόδους του, το τρέχον δείγμα του τριγωνικού κύματος αντιστρέφεται! Πού είναι όμως το δεύτερο κύμα και πώς ελέγχουμε το αν βρίσκεται σε κάποια από τις ημιπεριόδους του; Για λόγους απλότητας, θεωρήσαμε ότι οι τιμές του δεύτερου κύματος προκύπτουν από τη μεταβλητή phase, από την οποία προκύπτουν και οι τιμές του κύριου κύματος (του τριγωνικού) που παράγει το πρόγραμμα. Αλλά το δεύτερο κύμα πρέπει να έχει αρκετά μεγαλύτερη συχνότητα από το άλλο. Για το λόγο αυτό, όταν εξετάζουμε τη φάση του δεύτερου κύματος θεωρούμε ότι η μεταβλητή phase έχει μικρότερο εύρος τιμών (άρα η φάση του ολοκληρώνει ταχύτερα έναν κύκλο). Συγκεκριμένα, θεωρούμε ότι το πιο σημαντικό bit δεν είναι το όγδοο του δεύτερου byte, αλλά το έβδομο, το έκτο, το πέμπτο, το τέταρτο, το τρίτο, το δεύτερο ή το πρώτο. Ποιο από όλα θα θεωρήσουμε σαν το πιο σημαντικό καθορίζεται από τη μεταβλητή depth. Έτσι, η τιμή της depth καθορίζει το υποθετικό μέγεθος της phase και κατ' επέκταση τη συχνότητα του δεύτερου κύματος. Σκεφτείτε τώρα το εξής: Το πιο σημαντικό bit της μεταβλητής phase (όποιο κι αν έχουμε θεωρήσει ότι είναι) έχει την τιμή μηδέν για τα μισά βήματα της φάσης και την τιμή 1 για τα άλλα μισά. Με άλλα λόγια, το πιο σημαντικό bit δηλώνει σε ποια από τις δύο ημιπεριόδους του κύματος βρισκόμαστε: στη μία ή στην άλλη ;) Ο κώδικας, λοιπόν, ελέγχει την τιμή του bit το οποίο θεωρούμε ως πιο σημαντικό κι όταν έχει την τιμή 1 (όταν βρισκόμαστε στη μία από τις δύο ημιπεριόδους) αντιστρέφει το τρέχον δείγμα του τριγωνικού κύματος. Παρατηρείστε ότι αυτό το τμήμα του κώδικα εκτελείται πάντα! Η αντιστροφή του τριγωνικού κύματος μπορεί να συμβεί ανά πάσα στιγμή και δεν έχουμε κανένα λόγο να καθυστερούμε την εξέλιξη του συγκεκριμένου εφέ. Τέλος, οφείλουμε να παραδεχτούμε ότι η υλοποίησή μας είναι λίγο προβληματική. Βλέπετε, με το τέχνασμα που έχουμε επιστρατεύσει, η συχνότητα του δεύτερου κύματος αποτελεί πάντοτε ακέραιο πολλαπλάσιο (x2, x4, ..., x128) της συχνότητας του κύριου κύματος. Αυτός ο περιορισμός δίνει ελάχιστες επιλογές στο χρήστη και δεν επιτρέπει τη σωστή υλοποίηση του εφέ. Στην ιδανική περίπτωση, οι συχνότητες των κυμάτων που συμμετέχουν στο ring modulation πρέπει να *μην* σχετίζονται αρμονικά.

Ακούστε τα όλα!

Κάπου εδώ, η συγκεκριμένη σειρά άρθρων ολοκληρώνεται. Ελπίζουμε να ξυπνήσαμε όμορφα, νοσταλγικά συναισθήματα στους παλιότερους – αλλά και να προκαλέσαμε το ενδιαφέρον στους νεότερους. Σε κάθε περίπτωση, θέλουμε να πιστεύουμε ότι η παρουσίαση όλων αυτών των προγραμματιστικών τεχνικών άνοιξε νέους ορίζοντες στη σκέψη σας! Στο μέλλον ενδέχεται να επανέλθουμε στο θέμα. Μέχρι τότε θα σας αφήσουμε να παίξετε με τα προγράμματα που παρουσιάσαμε και τους ήχους που παράγουν. Κι αν αυτά δεν αρκούν για να πορωθείτε, θα σας προτείναμε να κάνετε μια βόλτα από το http://fonix.dyndns.org/soasc, όπου και θα βρείτε ηχογραφήσεις από τη μουσική όλων των παιχνιδιών του Commodore 64 ;)



Πλήρη **άρθρα που συμπληρώνουν** το περιοδικό. Το **video podcasts** του περιοδικού [deltaCast]. Τα προκλητικά **mind hacks**. Όλα τα βίντεο από τα **live streams** [deltaStream]. Ολόκληρα **blasts from the past**. Τα πάντα ενδιαφέροντα **fora**. Το σημείο συνάντησης για τους φίλους του περιοδικού.

deltahacker.gr

Αληθινό, διασκεδαστικό hacking **για όλους!**

Skill: Beginner

Tags: demoscene, cracktro, demo, Commodore 64, Amiga 500

DEMOSCENE

by GiannouG

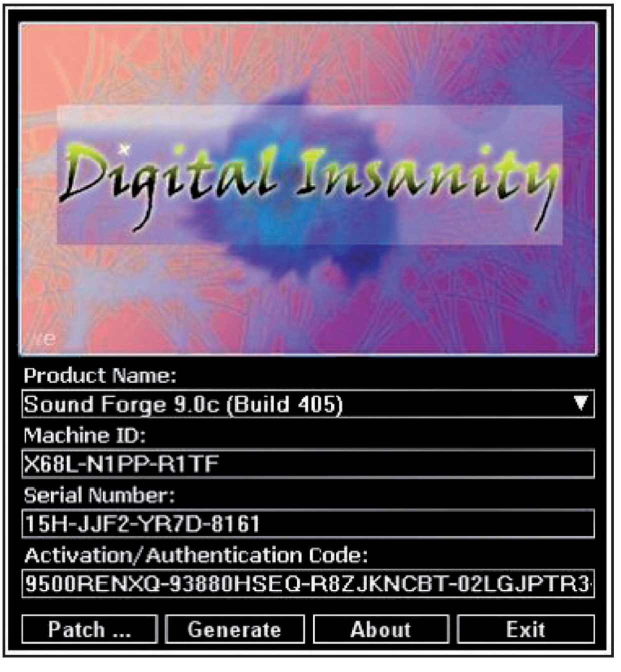


Αν και δεν μας αρέσουν τα τυχερά παιχνίδια —geeks είμαστε και ξέρουμε να εκτιμούμε τις πιθανότητες—, θα στοιχηματίζαμε άνετα ότι έχει πέσει στα χέρια σας *τουλάχιστον* μία φορά κάποιος key generator. Όχι ότι το επιδιώξατε, αλλά έτυχε. Πιθανώς λοιπόν να 'χετε παρατηρήσει ότι τα μικρά αυτά προγράμματα ενσωματώνουν συχνά και κάποιο demo. Και κάπως έτσι, εντελώς τυχαία, όλοι μας έχουμε έρθει σε επαφή με τη λεγόμενη demoscene!

watch it in HD: http://www.youtube.com/watch?v=D0igUqzGpTk

Τα διάφορα key generators ή αλλιώς keygens συνοδεύονται από ηλεκτρονική μουσική και προβάλλουν διάφορα γραφικά. Δεν είναι πάντα κάτι το φανταχτερό, ούτε κάτι που θα μας αφήσει με το στόμα ανοιχτό. Απλά, πρόκειται για την υπογραφή της ομάδας που κατάφερε να σπάσει την εφαρμογή για την οποία γράφτηκε το keygen. Αν λοιπόν έχετε δει έστω και μια φορά ένα τέτοιο πρόγραμμα, τότε έχετε πάρει ήδη μια γεύση από τη λεγόμενη demoscene! Μαζί με την επίδειξη δύναμης μιας ομάδας, η οποία καταφέρνει και α) βρίσκει κάποιο πρόγραμμα και β) το σπάζει (σε χρόνο ρεκόρ), συχνά γίνεται κι επίδειξη των καλλιτεχνικών δυνατοτήτων. Έτσι, οι μεγάλες ομάδες cracking απαρτίζονται από πολλά άτομα, που θα μπορούσαμε να τα χωρίσουμε σε δύο υποομάδες: Σε αυτούς που προσπερνούν τις δικλείδες ασφαλείας των προγραμμάτων και σε εκείνους που δημιουργούν τα γραφικά και τη μουσική, τα οποία συνοδεύουν τα προγράμματα της ομάδας. Φυσικά, ο διαχωρισμός αυτός δεν είναι απόλυτος. Υπάρχουν και μέλη αυτών των ομάδων που καταπιάνονται με αμφότερες τις εργασίες. Όπως καταλαβαίνετε, μιλάμε για προγραμματιστές με μεράκι και μεγάλο ταλέντο!

Ένα keygen δημιουργεί κλειδιά για την εγκατάσταση κάποιου σπασμένου software. Ένα keygen προβάλει συνήθως μερικά γραφικά με το όνομα της ομάδας που το δημιούργησε, καθώς και κάποιο τραγούδι που θυμίζει την εποχή των 8bit computers!



Πειρατεία με καλλιτεχνική φλέβα

Θα ήταν λάθος να χαρακτηρίσουμε ως «πειρατές» όλους όσοι ασχολούνται με τη demoscene. Η ιστορία με αυτή τη σκηνή ξεκινά πριν από μερικές δεκαετίες, τότε που το Internet δεν υπήρχε και τα προγράμματα αποθηκεύονταν σε κασέτες ήχου ή, στην καλύτερη περίπτωση, σε μεγάλες εύκαμπτες δισκέτες (5 1/4 ιντσών). Τα εμπορικά προγράμματα της εποχής ενσωμάτωναν ιδιαίτερα έξυπνους μηχανισμούς για την προστασία από την παράνομη αντιγραφή. Ωστόσο, όπως και σήμερα, έτσι και τότε υπήρχαν τα καλά παιδιά που πείραζαν τα προγράμματα και αφαιρούσαν πλήρως την προστασία ή έβρισκαν τρόπους να την παρακάμψουν. Σχεδόν πάντα, όμως, μαζί με το σπάσιμο του προγράμματος φύτευαν κι ένα κομμάτι κώδικα, που έτρεχε *πριν* ξεκινήσει το κυρίως πρόγραμμα. Αυτό το κομμάτι κώδικα παρήγαγε φανταχτερά γραφικά, εμφάνιζε μηνύματα που κινούνταν σε εντυπωσιακές τροχιές, έπαιζε μουσική κ.λπ. Η ονομασία αυτών των μικρών προγραμμάτων ήταν cracktro!

Κάποια στιγμή, τα προγράμματα που είχαν ως στόχο να εντυπωσιάσουν το κοινό χωρίστηκαν από εκείνα που προσπερνούσαν τα διάφορα κλειδώματα. Έτσι, άρχισαν να εμφανίζονται ολόκληρα προγράμματα από τους προγραμματιστές των διαφόρων ομάδων, τα οποία επιδείκνυαν το ταλέντο, τις προγραμματιστικές ικανότητες και τη βαθιά γνώση του hardware. Εδώ που τα λέμε, ακόμη και μια απλή σφαίρα που χοροπηδούσε στην

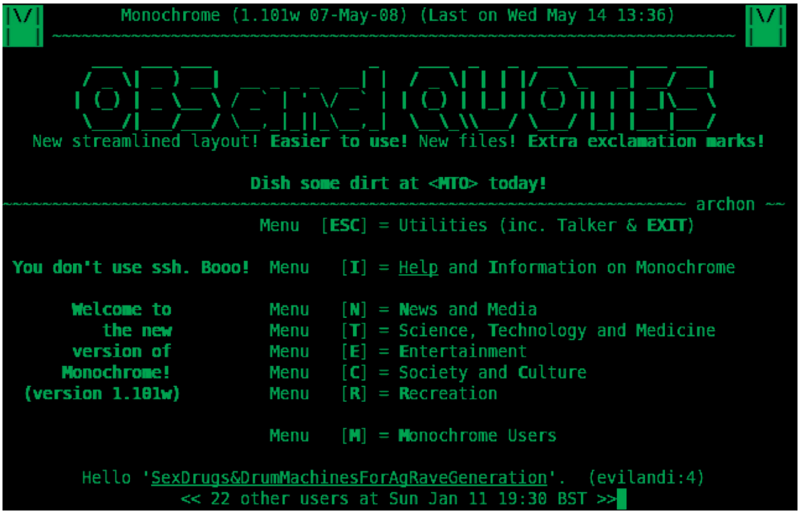
οθόνη ήταν τρομερό κατόρθωμα για τα μηχανήματα της εποχής. Σήμερα μπορούμε να δημιουργήσουμε παρόμοια γραφικά σε μεγαλύτερες αναλύσεις, με περισσότερα χρώματα και πολλά εφέ, γράφοντας ελάχιστες γραμμές κώδικα και χωρίς να έχουμε τις παραμικρές γνώσεις για το hardware του υπολογιστή. Τότε όμως τα πράγματα ήταν εντελώς διαφορετικά. Τα γραφικά, τα κινούμενα μηνύματα και η μουσική παράγονταν από κώδικα σε γλώσσα μηχανής! Τα προγράμματα που πετύχαιναν όλα τα παραπάνω ονομάζονταν demo και η αντίστοιχη κοινότητα των προγραμματιστών ονομάστηκε demoscene. Εδώ πρέπει να διευκρινίσουμε ότι για να θεωρηθεί ένα πρόγραμμα demo, πρέπει τα γραφικά και η μουσική να παράγονται σε πραγματικό χρόνο (realtime) και να μην είναι προσχεδιασμένα (pre-rendered). Μάλιστα, αξίζει να σημειώσουμε ότι υπάρχουν αρκετές κατηγορίες προγραμμάτων demo, καθεμία από τις οποίες επιβάλει διαφορετικούς κανόνες για το πώς παράγονται τα εφέ, ποια γλώσσα προγραμματισμού χρησιμοποιείται, τι μέγεθος οφείλει να έχει το τελικό πρόγραμμα κ.ο.κ.



Όλα ξεκίνησαν από τα cracktro, στις αρχές του '80. Οι διάφορες ομάδες που έσπαγαν τα προγράμματα ήθελαν να αφήσουν την υπογραφή τους και φυσικά να προκαλέσουν τις άλλες ομάδες ;)

Από τα cracktro στα demoparty

Στα τέλη της δεκαετίας του '80 το Internet βρισκόταν σε εμβρυακό στάδιο ανάπτυξης, ενώ ήταν προσβάσιμο από ελάχιστους χρήστες. Εκείνη την εποχή άνθιζαν τα λεγόμενα BBSes: Υπολογιστές στους οποίους συνδέονταν διάφοροι χρήστες μέσω modem κι αντάλλασσαν προγράμματα και μηνύματα. Ωστόσο, οι χαμηλές ταχύτητες σύνδεσης, το υψηλό κόστος (οι συνδέσεις είχαν χρονοχρέωση, όπως οι φωνητικές κλήσεις) και το γεγονός ότι δεν υπήρχαν παντού BBSes, καθιστούσαν αυτόν τον τρόπο δικτύωσης δύσκολο κι ασύμφορο. Γι' αυτούς τους λόγους, οι ενδιαφερόμενοι οργάνωναν μεγάλες συγκεντρώσεις ώστε να γνωριστούν καλύτερα και να ανταλλάξουν πληροφορίες, προγράμματα και ιδέες! Κάπως έτσι ξεκίνησαν τα λεγόμενα copyparty, στα οποία κυριαρχούσε η παράνομη αντιγραφή λογισμικού! Τα copyparty κίνησαν αμέσως το ενδιαφέρον της αστυνομίας και σταδιακά άρχισαν να παίρνουν κακή φήμη. Έτσι, οι φίλοι



Τα BBSes, οι πρόγονοι των σημερινών forums, ήταν οι κύριοι κόμβοι για την ανταλλαγή των demos.

των υπολογιστών και του προγραμματισμού άρχισαν σιγά-σιγά να διοργανώνουν τα λεγόμενα demoparty. Σε αυτές τις συγκεντρώσεις πραγματοποιούνταν διαγωνισμοί που περιελάμβαναν την κατασκευή προγραμμάτων demo, αλλά μη νομίζετε ότι η πειρατεία απουσίαζε πλήρως ;)

Τα περισσότερα demoparty οργανώνονταν κυρίως στην Ευρώπη: Στη Γερμανία, στη Δανία, στην Ιταλία, στη Γαλλία αλλά κι εδώ, στην Ελλάδα! Αυτές οι συγκεντρώσεις διαρκούσαν από δύο έως πέντε μέρες και πραγματοποιούνταν είτε σε χώρους κάποιου Πανεπιστημίου είτε και σε γήπεδα! Οι διοργανωτές παρείχαν πάντα κάποιον χώρο για ύπνο, φαγητό κι άφθονο αλκοόλ. Το μόνο που έπρεπε να φέρουν οι καλεσμένοι ήταν κάποιο sleeping bag και φυσικά τον υπολογιστή τους. Αυτό όμως ήταν αρκετά δύσκολο εκείνη την εποχή, αφού δεν υπήρχαν ακόμα τα laptops. Αλήθεια, ξέρετε σε τι μηχανήματα δούλευαν τότε;

Τα πρώτα demo

Τα περισσότερα και πιο εντυπωσιακά demos γράφτηκαν για τον Commodore 64 και την Amiga 500. O Commodore 64 ήταν από τους πρώτους *φθηνούς* υπολογιστές για οικιακή χρήση, ο οποίος διέθετε αρκετά ανεπτυγμένο hardware για γραφικά και ήχο. Η δημοτικότητα και οι δυνατότητές του τον κατέστησαν κυρίαρχο τόσο στο χώρο των οικιακών υπολογιστών, όσο και στο χώρο της demoscene. Στα τέλη της δεκαετίας του '80 έκαναν την εμφάνισή τους κι ορισμένες ομάδες cracking, που συνεχίζουν το έργο τους μέχρι σήμερα! Μερικές από αυτές είναι η Razor 1911 (RZT), η FairLight (FLT) και η Tristar & Red Sector Incorporated (TRSI), που τις συναντάμε πολύ συχνά στους torrent tracker. Ε, λοιπόν, όλες αυτές οι ομάδες ξεκίνησαν φτιάχνοντας demos και cracks για διάφορα παιχνίδια του Commodore 64. Μερικές αξιόλογες δημιουργίες της εποχής είναι το Edge of Disgrace των Booze Design, το Deus Ex Machina από Crest & Oxyron (ιδιαίτερα ενδιαφέρον στο τέλος του πρώτου μέρους, που ζητάει να γυρίσεις την δισκέτα για να συνεχίσει) και το Natural Wonders 2 από Oxyron. Ο ευκολότερος τρόπος για να τις δούμε είναι να ψάξουμε στο YouTube, αλλά όσοι έχουν όρεξη θα μπορούσαν να ψάξουν τα ίδια τα προγράμματα και να τα τρέξουν σε κάποιον emulator!



Assembly 2002! Στα Assembly demoparty μαζεύεται πάντα ο περισσότερος κόσμος. Αξίζει να αναφέρουμε ότι υπήρξαν φορές που οι διοργανωτές δυσκολεύτηκαν να νοικιάσουν τον απαιτούμενο χώρο!

Καθώς προχώρησε η τεχνολογία και βγήκαν νέα συστήματα, η demoscene έστρεψε το ενδιαφέρον της σε πιο ισχυρές πλατφόρμες. Μεταξύ αυτών ήταν και η Amiga 500, ένας υπολογιστής που ακολουθούσε τα χνάρια του Commodore 64. Ο επεξεργαστής της Amiga ήταν 32bit αντί για 8bit, η ανάλυση και τα χρώματα είχαν αυξηθεί, ενώ και το κύκλωμα ήχου είχε αποκτήσει περισσότερες δυνατότητες. Σιγά-σιγά τα BBSes άρχισαν να γεμίζουν με demos για την Amiga, ενώ τα διάφορα εμπορικά προγράμματα έσπαγαν μέσα σε λίγες μέρες! Φυσικά, αυτή η στροφή στο ενδιαφέρον των προγραμματιστών εκδηλώθηκε και στα αντίστοιχα party, στα οποία άρχισε να κυριαρχεί η Amiga 500. Ωστόσο, πρέπει να πούμε ότι ο Commodore 64 δεν έλειψε ποτέ από αυτές τις εκδηλώσεις!

Μερικά ακόμα συστήματα της ίδιας εποχής είναι ο Atari 800, ο Atari ST, ο Apple IIGS, ο Commodore VIC-20 αλλά και ο ZX Spectrum. Τα συγκεκριμένα μηχανήματα προσέφεραν συγκριτικά περιορισμένες δυνατότητες στον τομέα των γραφικών και ήχου. Ως εκ τούτου, αν και υπήρξαν πολλές ομάδες που ασχολήθηκαν μαζί τους, ποτέ δεν απέκτησαν την αίγλη και τη φήμη των άλλων δύο.

Η demoscene σήμερα

Από τις αρχές του 1990 μέχρι σήμερα δεν έχουν αλλάξει και πολλά στο χώρο της demoscene. Μπορεί τώρα να υπάρχουν ισχυρότερα μηχανήματα, αλλά οι σκληροπυρηνικοί προγραμματιστές ασχολούνται αποκλειστικά με το παλιό hardware! Τα περισσότερα group έχουν περάσει στα PC και στην πλατφόρμα των Windows, αλλά υπάρχουν αρκετές ομάδες που κυκλοφορούν demos για τον Commodore 64! Πάντως, εκτός από τα PC, την τιμητική τους έχουν και πολλές ακόμα πλατφόρμες. Ορισμένες ομάδες έχουν στραφεί στις κονσόλες και υπάρχουν πολλά demos για το Playstation, το Xbox αλλά και το Wii. Demos υπάρχουν ακόμα και για φορητές συσκευές, όπως το Gameboy Classic, το Gameboy Color αλλά και το PSP. Φυσικά στόχος έχουν γίνει και τα κινητά, τα PDAs και τα κομπιουτεράκια τσέπης!



Το Commodore 64 ήταν ένα από τα μηχανήματα που χάραξε την πορεία της demoscene. Οι απίστευτες δυνατότητές του για τα δεδομένα της εποχής επέτρεπαν στους demoers να δημιουργούν εντυπωσιακά οπτικά και ηχητικά εφέ!



Η Amiga 500 κατάφερε να γίνει ο διάδοχος του Commodore 64. Προσέφερε καλύτερα γραφικά και ήχο, ενώ κι ο επεξεργαστής της ήταν πολύ πιο ισχυρός!



Η ομάδα ASD απαρτίζεται από Έλληνες και έχει κερδίσει την πρώτη θέση σε αρκετούς διαγωνισμούς. Το screenshots προέρχεται από το demo Lifeforce, που κέρδισε την πρώτη θέση στο Assembly 07!

Παρουσία στην Ελλάδα

Αμέ, γιατί όχι; Φυσικά κι έχει γίνει demoparty στην Ελλάδα! Εξάλλου, υπάρχουν και demo groups που αποτελούνται από Έλληνες! Μείνατε με το στόμα ανοιχτό; Gardening '95, '96 και '97, Digital Nexus 2001 και ReAct, είναι μερικά από τα demoparty που διοργανώθηκαν στην Ελλάδα. Δυστυχώς, η διεξαγωγή τους σταμάτησε εξαιτίας του μικρού ενδιαφέροντος από πλευράς κοινού. Το τελευταίο demoparty στην Ελλάδα ήταν το PixelShow (2007). Από τότε δεν έχουμε δει κάποια νεότερη προσπάθεια και το πιθανότερο είναι να μην ξαναδούμε. Μερικά ελληνικά groups με πρόσφατες κυκλοφορίες αλλά και μεγάλο παρελθόν, είναι τα Andromeda Software Development (ASD), Dirty Minds και Optimus. Αξίζει να σημειωθεί πως τα εν λόγω groups έχουν αρκετές πρωτιές σε πολλά demoparty του εξωτερικού. Για παράδειγμα, οι ASD κατέκτησαν δύο φορές την πρωτιά στο Assembly 2007 και στο Assembly 2011. Πρόκειται για demoparty που διοργανώνεται αδιάκοπα από το 1992 και συγκεντρώνει κάθε χρόνο αρκετές χιλιάδες επισκέπτες!

Κλείνοντας

Θα μπορούσε να πει κανείς ότι η demoscene αποτελεί ένα είδος τέχνης. Για να δημιουργήσεις κάτι χρησιμοποιείς τον υπολογιστή, αλλά οι γνώσεις προγραμματισμού δεν αρκούν. Χρειάζεται και αρκετό ταλέντο, όπως επίσης γνώσεις μουσικής και σχεδίου. Όπως καταλαβαίνετε, όλα ξεκίνησαν από την πειρατεία και την ανάγκη των χρηστών για αναγνώριση και φήμη. Από τα απλά cracktro που στόλιζαν τις εισαγωγές των παιχνιδιών, καταλήξαμε σε συγκεντρώσεις χιλιάδων ανθρώπων, που δουλεύουν ομαδικά και ανταλλάσουν γνώσεις και εμπειρία...



Είμαστε κοινωνικοί — ειδικά στον online κόσμο.
Είμαστε αστείοι, ευφυείς και φινετσάτοι — ειδικά στον online κόσμο.
Λέγεται ότι έχουμε καβαλήσει το καλάμι — τόσο στον online, όσο και στον offline κόσμο.
Όπως και να 'χει, το διαχειριζόμαστε γενναία.
Περάστε από τα στέκια μας να τα πούμε — ή να μας την πείτε.
twitter.com/deltaHacker
facebook.com/deltaHacker

Skill: Beginner

Tags: Google Dorks, advanced searching, privacy

Πλαστοπροσωπία και κατασκοπεία



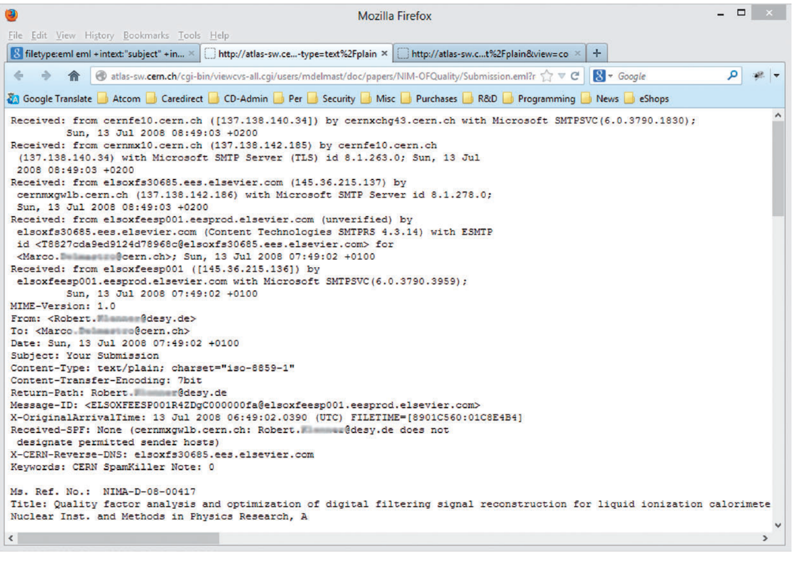
Τα τελευταία χρόνια έχουμε γίνει μάρτυρες μιας παγκόσμιας επανάστασης, που έφερε η ιντερνετική εποχή. Αυτή, μπορεί να συγκριθεί μόνο με τη βιομηχανική ή με αυτή του διαφωτισμού! Μην σας φαίνεται περίεργο, ούτε υπερβολικό. Για να αναλογιστείτε το μέγεθος αυτής της επανάστασης, σκεφτείτε, ας πούμε, ότι ο διαφωτισμός αφορούσε μόνο στην Ευρώπη και όχι σ' ολόκληρο τον πλανήτη, όπως τώρα. Υπαναχωρούμε λοιπόν κι ανασκευάζουμε την αρχική μας δήλωση. Για να είμαστε πιο σωστοί, πρέπει να πούμε ότι η ιντερνετική επανάσταση συγκρίνεται *μόνο* με αυτή της ανακάλυψης του τροχού. Με τίποτα λιγότερο.

by Thiseas

Πολύ ωραία όλα αυτά, τέλεια και καταπληκτικά. Αν μάλιστα ο κόσμος μας ήταν γεμάτος από ανθρώπους με καλή πίστη ή χωρίς καμιά πονηρία, δεν θα είχαμε κανένα πρόβλημα. Τα πράγματα όμως δεν είναι έτσι. Κάθε νέα εξέλιξη επιφέρει και κάποιες παράπλευρες απώλειες ή έστω νέα βάσανα. Υπάρχουν περιπτώσεις που κάποιος, με καμουφλάζ ένα πληκτρολόγιο και το δίκτυο, μπορεί να γίνει πάρα πολύ επικίνδυνος, τόσο για τον καθένα ξεχωριστά όσο και για το κοινωνικό σύνολο. Τα προφανή ερωτήματα που ανακύπτουν τώρα είναι τα εξής: Είναι πολλές αυτές οι περιπτώσεις; Πόσο εύκολο είναι για κάποιον να κάνει κάτι επικίνδυνο για εμένα ή για όλους εμάς; Ελάτε να μελετήσουμε μαζί αυτά τα ερωτήματα. Θα σας παραθέσουμε μερικά γεγονότα και θα αφήσουμε τις απαντήσεις σε εσάς.

Google: Ο καλύτερός μας φίλος

Είναι γνωστό ότι ο mister Google είναι ο καλύτερός μας φίλος. Χωρίς αυτόν (και τους ομοίους του), ο τεράστιος όγκος πληροφοριών του διαδικτύου θα ήταν απλά άχρηστος. Είναι ένα εργαλείο τρομακτικά χρήσιμο αλλά και τρομακτικά αιχμηρό. Ως τέτοιο, αν το χρησιμοποιήσουμε προσεκτικά —ή καλύτερα αν το χρησιμοποιήσουμε με συγκεκριμένο τρόπο— μπορούμε να πετύχουμε αξιοσημείωτα αποτελέσματα! Ναι, καλά το καταλάβατε: Η κουβέντα πηγαίνει στα περίφημα Google Dorks (www.exploit-db.com/google-dorks). Πρόκειται για λέξεις-κλειδιά ή και τελεστές, που μπορούμε να εισάγουμε σε μια αναζήτηση ώστε να πάρουμε στοχευμένα (μα πολύ στοχευμένα) αποτελέσματα. Οι αναζητήσεις, όμως, είναι σαν τον έρωτα: Θέλουν δύο για να λειτουργήσουν σωστά! Στην περίπτωσή μας, το έτερον ήμισυ είναι τα αποθηκευμένα δεδομένα. Μπορεί να κάνουμε την τέλεια αναζήτηση χρησιμοποιώντας πανέξυπνα τους πιο εξεζητημένους τελεστές, αλλά ως αποτέλεσμα να μην πάρουμε τίποτα αν *δεν* υπάρχουν δεδομένα. Όμως, τι είδους δεδομένα μπορούμε να βρούμε στο Internet; Θα μπορούσε κανείς να πει «τα πάντα», σκεπτόμενος λίγο βιαστικά. Εμείς όμως θα απαντούσαμε ως εξής: «Τα πάντα. Ακόμη κι αυτά που δεν θα έπρεπε να βρίσκουμε». Δεν μένει παρά να το αποδείξουμε.



Εικόνα 1
Το πρώτο θύμα της αναζήτησης είναι ο server ονόματι Atlas, στο Cern. Πολύ χύμα, έτσι;

```
To submit a revision, please go to http://ees.elsevier.com/nima/ and click
"login" underneath the journal title banner.  You may then type in your user
name/password and click "Author Login."

Your username is: [redacted]
Your password is: [redacted]

On your Main Menu page is a folder entitled "Submissions Needing Revision".
You will find your submission record there.  Also, the reviewer(s) may have
uploaded detailed comments on your manuscript. Click on the "Submissions
Needing Revision" from your main menu, then click on "View Reviewer
Attachments" to access any detailed comments from the reviewer(s) that may have
been included.

With best regards,

Robert [redacted]
Editor
Nuclear Inst. and Methods in Physics Research, A
```

Εικόνα 2
Πυρηνικό εργαστήριο, με κωδικό και password!

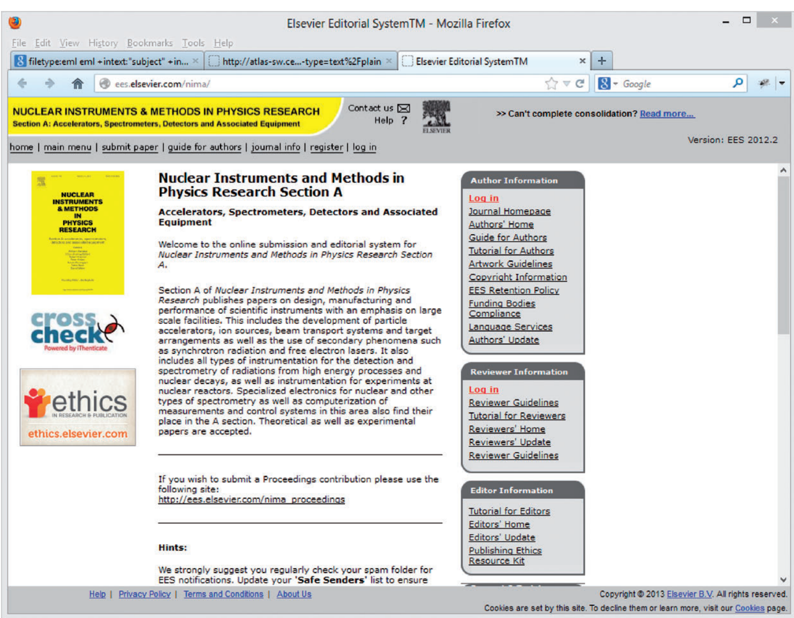
Google Dorks

Για όσους δεν τα γνωρίζουν, πρόκειται για τελεστές που μπορούν να χρησιμοποιηθούν ώστε να εξειδικεύσουν μια αναζήτηση. Για να μπούμε κατευθείαν στο θέμα, θα σας δείξουμε το πρώτο Dork που φτιάξαμε:

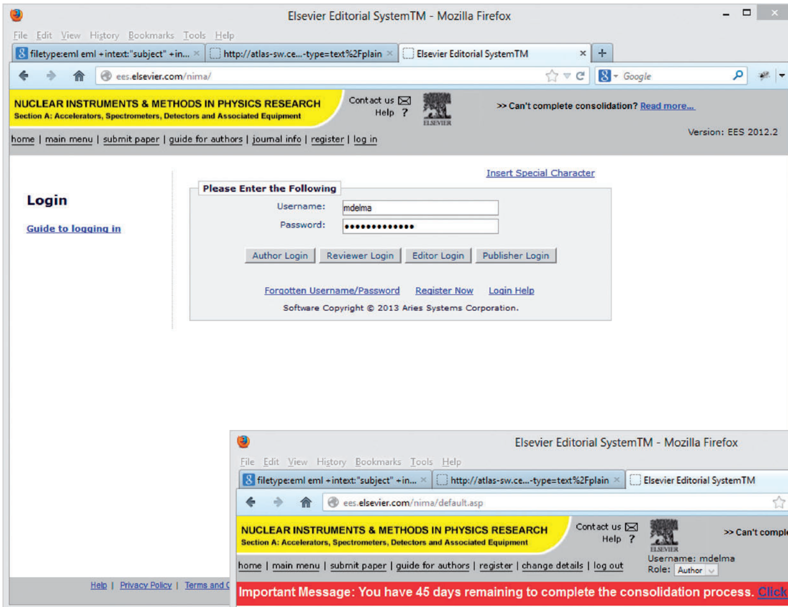
filetype:eml **+intext**:Subject **+intext**:From **+intext**:To **+intext**:user **+intext**:password **+inurl**:cern

Το παραπάνω θα το δώσουμε στη μηχανή αναζήτησης της Google και θ' αφήσουμε τα υπόλοιπα σε αυτήν. Οι τελεστές που χρησιμοποιήσαμε στο συγκεκριμένο dork είναι οι ακόλουθοι:

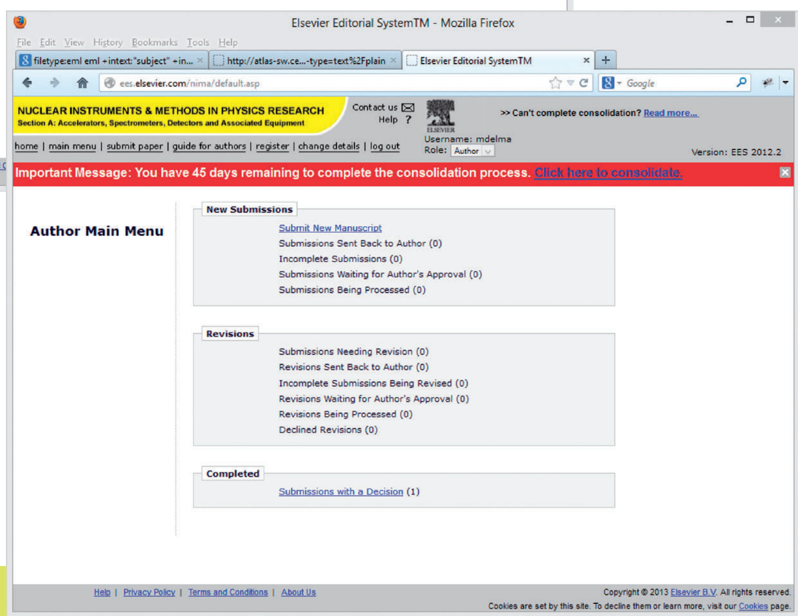
- **filetype:** Ζητάμε να ψάξει μόνο σε αρχεία που έχουν συγκεκριμένη προέκταση. Στην περίπτωση μας, ζητάμε την προέκταση eml. Ξέρετε ποια αρχεία έχουν τέτοια προέκταση; Αυτά που περιέχουν δε-



Εικόνα 3
Το site του πυρηνικού εργαστηρίου.



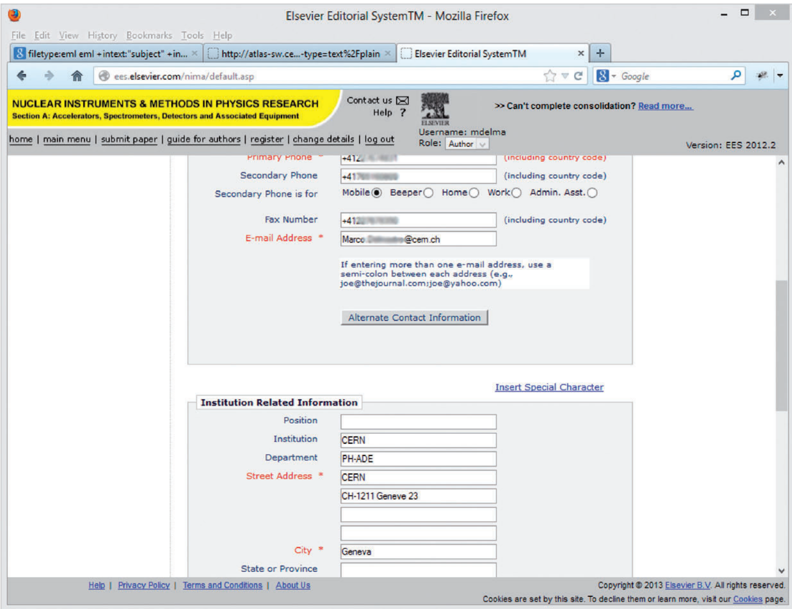
Εικόνα 4
Για να δούμε, τι θα δούμε... ;)



Εικόνα 5
I 'm in, baby!

δομένα από emails. Χρησιμοποιούνται από αρκετούς email clients κι ένας από αυτούς είναι και το γνωστό και μη εξαιρετέο Microsoft Outlook.

- **intext:** Το χρησιμοποιούμε για να πραγματοποιηθεί μια αναζήτηση *εντός* των σχετικών αρχείων. Όπως βλέπετε, το dork κάνει ευρεία χρήση του συγκεκριμένου τελεστή.
- **+**: Αυτός ο τελεστής λειτουργεί όπως ο λογικός τελεστής AND. Με τη χρήση του απαιτούμε να ικανοποιούνται *και* οι συνθήκες που τον ακολουθούν. Για παράδειγμα, το κομμάτι Dork +intext:»Subject» +intext:»From» σημαίνει οτι θέλουμε τα αρχεία που περιέχουν τη λέξη Subject αλλά *και* τη λέξη From.
- **inurl:** Με τη χρήση αυτού του τελεστή, η αναζήτηση θα επιστρέφει αρχεία που το URL τους περιέχει τη λέξη «cern».

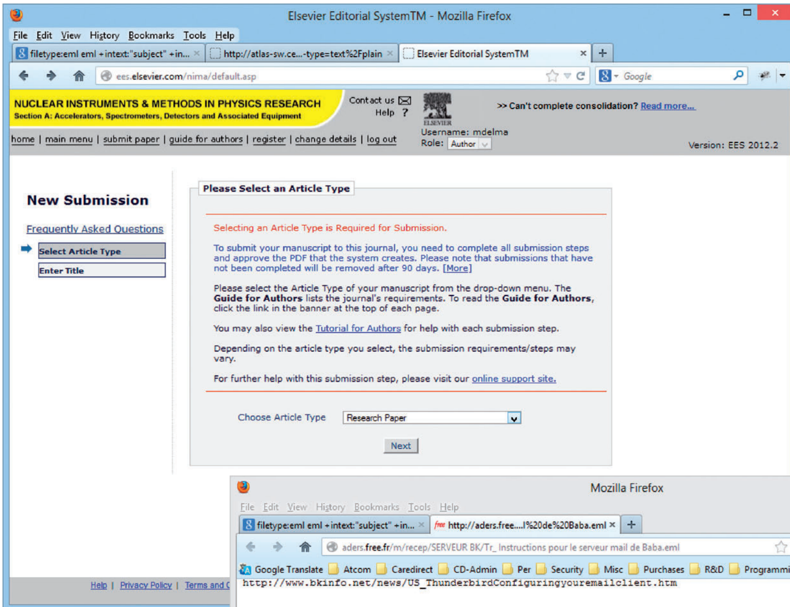


Εικόνα 6
Hello sir... I am YOU!

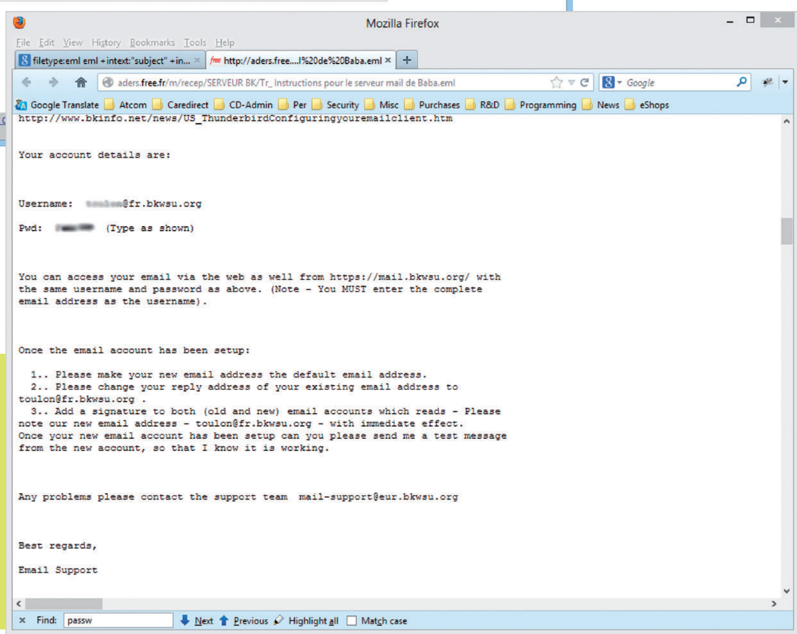
Ευρήματα

Δίνοντας στο Google το παραπάνω dork πήραμε μια λίστα με κάποια πιθανά αποτελέσματα. Μερικά είναι αρκετά διαφωτιστικά, θα λέγαμε (βλ. εικόνα 1). Μεταξύ άλλων βρήκαμε ένα email. Ένα email που κανονικά δεν θα έπρεπε να είναι προσβάσιμο από οποιονδήποτε (πόσο μάλλον από το Google). Διαβάζοντας το συγκεκριμένο μήνυμα αποκτάμε πληροφορίες για τις διευθύνσεις email ορισμένων άλλων ατόμων (επιστημόνων, στη συγκεκριμένη περίπτωση), μαθαίνουμε για διευθύνσεις IP (εσωτερικές και εξωτερικές), για υπηρεσίες που τρέχουν σε αυτές τις διευθύνσεις (όπως η υπηρεσία Microsoft SMTPSVC) και πολλά άλλα. Αν σχεδιάζαμε μια αληθινή επίθεση, θα είχαμε κάνει ένα πολύ καλό πρώτο βήμα. Συνεχίζοντας το διάβασμα πέφτει το μάτι μας πάνω σε κάτι πολύ ενδιαφέρον: στη λέξη «password» (βλ. εικόνα 2). Το site για το οποίο αναφέρεται το password φαίνεται στην εικόνα 3 και καταλαβαίνουμε αμέσως ότι έχει επιστημονικό περιεχόμενο, με δημοσιεύσεις πυρηνικής φυσικής κι όχι μόνο. Μπορούμε άραγε να γίνουμε κι εμείς πυρηνικοί επιστήμονες και να υποβάλλουμε (submit) το πρώτο μας paper; Αυτό που έχουμε να κάνουμε είναι να δοκιμάσουμε τα credentials που βρήκαμε στο email, για να δούμε αν μπορούμε να συνδεθούμε ως νόμιμοι χρήστες (βλ. εικόνα 4). Όπως βλέπετε στην εικόνα 5, η προσπάθειά μας ήταν επιτυχημένη!

Ήταν τελικά τόσο εύκολο; Μάλλον ναι — και χρειαστήκαμε λιγότερο από 5 λεπτά! Μετά από όσα είδαμε, θα μπορούσε κανείς να συνεχίσει το ψάξιμο (ή την κατασκοπεία) εντός του site. Για παράδειγμα, με μια σύντομη αναζήτηση μπορούμε να δούμε τα προσωπικά στοιχεία του επιστήμονα τον οποίο υποδυόμαστε (βλ. εικόνα 6). Μπορούμε ακόμα να στείλουμε για δημοσίευση το πρώτο μας άρθρο, ως πυρηνικοί φυσικοί (βλ. εικόνα 7). Η αλήθεια είναι ότι κάνοντας αυτές τις δοκιμές μπήκαμε σε μεγάλο πειρασμό να στείλουμε το άρθρο που διαβάζετε, στη θέση κάποιου paper. Ωστόσο η λογική επικράτησε και δεν κάναμε καμιά κουταμάρα.



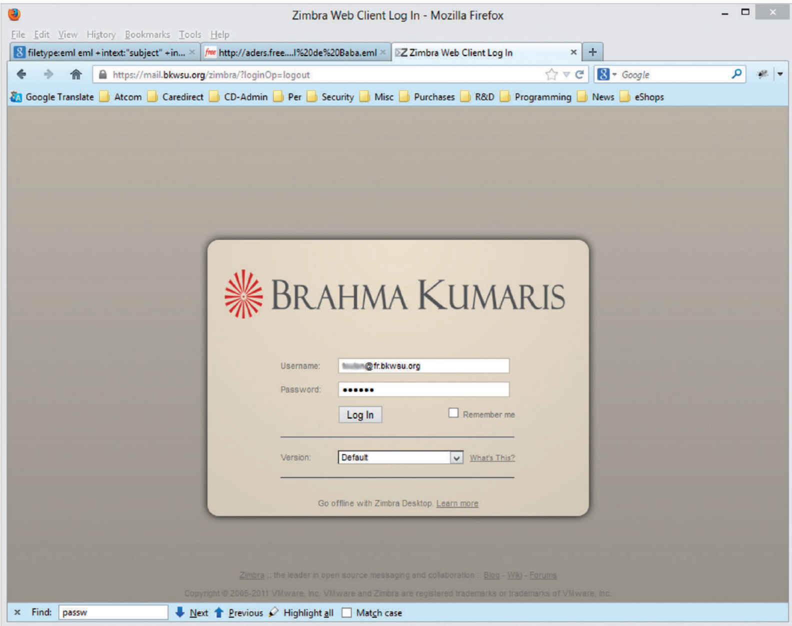
Εικόνα 7
Ενδιαφέρεστε να στείλετε άρθρο στο Cern; Θέλετε να κάνετε καριέρα σαν πυρηνικός φυσικός; Εμείς είμαστε εδώ!



Εικόνα 8
Θέλετε να κάνετε email cracking και να μπείτε παράνομα στο λογαριασμό κάποιου χρήστη; Piece of cake! (Σ.τ.Ε. Και τι δεν θα 'δινα για να 'γραφε κάποιος —οποιοσδήποτε— «κομμάτι από κέικ»! :S)

Ευρήματα ...more!

Ας αφήσουμε το CERN και τα ερευνητικά ιδρύματα κι ας δούμε τι άλλο μπορούμε να βρούμε με τα dorks. Τι θα λέγατε να κατασκοπεύσουμε την προσωπική ζωή κάποιου, χωρίς να μας πάρει ποτέ είδηση και χρησιμοποιώντας αποκλειστικά τις πληροφορίες που δίνει η μηχανή της Google; Με μια μικρή τροποποίηση του αρχικού dork μπορούμε να καταλήξουμε να διαβάζουμε ένα email, σαν αυτό της εικόνας 8. Όπως βλέπετε, έχουμε το email και το password κάποιου καημένου χρήστη, που κάποιος άλλος καημένος (admin) άφησε μέσα σ' έναν κατάλογο προσβάσιμο από το web! Για να δούμε, όμως: είναι πραγματικά τα στοιχεία που παρέχονται; Δεν έχουμε παρά να κάνουμε μια δοκιμή (βλ. εικόνα 9). Για μία ακόμα φορά, τα αποτελέσματα είναι απολύτως ικανοποιητικά (βλ. εικόνα 10). Είμαστε



Εικόνα 9
Για να δούμε, τι θα δούμε… (more)

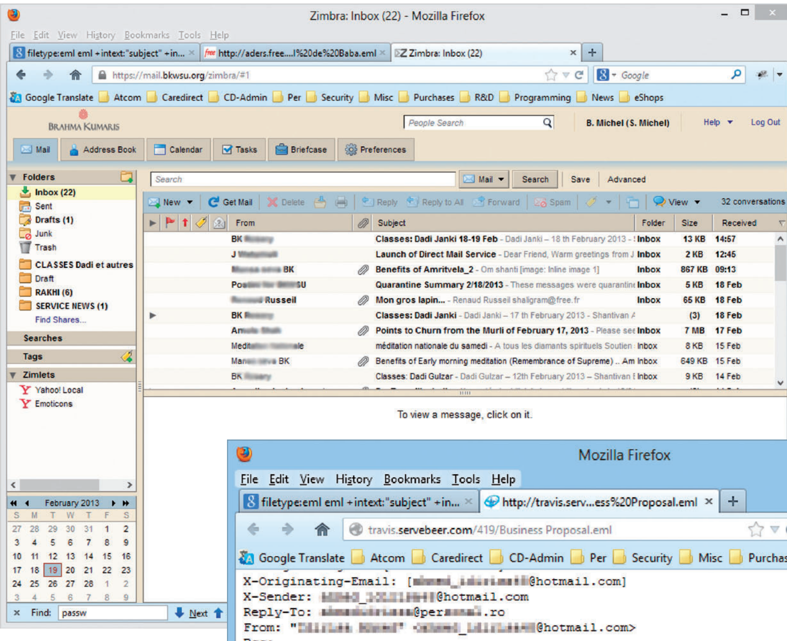
μέσα στο webmail ενός άμοιρου χρήστη, χωρίς να κουραστούμε καθόλου και χωρίς να παραβιάσουμε καμιά ασφάλεια, κανενός server. Ερώτηση κρίσεως: Είναι νόμιμο; Ναι! Είναι ηθικό: Όχι! Συμπέρασμα: Ότι είναι νόμιμο δεν είναι κατ' ανάγκη και ηθικό![1]

Για να μην ξεχνιόμαστε, να αναφέρουμε ότι δεν είναι καθόλου μα καθόλου ακίνδυνο, να έχουμε πρόσβαση στο email κάποιου. Μπορούμε να επέμβουμε ποικιλοτρόπως στην προσωπική και στην οικονομική του ζωή. Για παράδειγμα, μπορούμε να βρούμε λογαριασμούς PayPal. Η εν λόγω υπηρεσία δεν στέλνει ποτέ τα credentials μέσα στα emails, αλλά αυτό δεν μας ενοχλεί καθόλου. Μπορούμε να αποκτήσουμε πρόσβαση στο σχετικό λογαριασμό PayPal δηλώνοντας ότι ξεχάσαμε το password μας και περιμένοντας να φτάσει στο email το link για την εκ νέου ενεργοποίηση του λογαριασμού. Επίσης, να αναφέρουμε πως ό,τι έχουμε παραβιάσει μέχρι τώρα (όπως μπορείτε να δείτε και στις εικόνες) είναι σε επίπεδο HTTPS! Για να το πούμε αλλιώς, η παραβίαση έγινε σε υπηρεσίες που χρησιμοποιούν κρυπτογραφημένες συνδέσεις. Έτσι, μαζί με την ιδιωτικότητα του νόμιμου ιδιοκτήτη, προστατεύεται και η ιδιωτικότητα του κλέφτη ;)

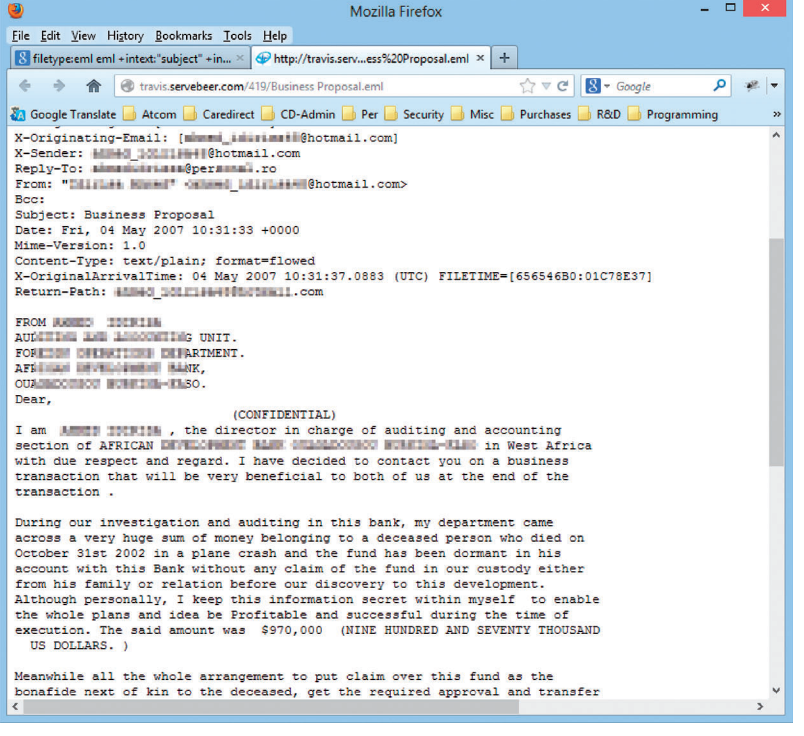
Ευρήματα ...and then some!

Υπάρχουν κι άλλες πληροφορίες που μπορούμε να υποκλέψουμε με τη βοήθεια του Google, όπως είναι οι μυστικές οικονομικές συναλλαγές (εικόνα 11). Σε αυτό το email ο κος Idirisa Ahmed (γνωστός τραπεζίτης!) δηλώνει στον παραλήπτη ότι είναι ιδιοκτήτης μιας μεγάλης τράπεζας και ότι έχει εντοπίσει έναν λογαριασμό με 970.000$, του οποίου ο ιδιοκτήτης είναι νεκρός. Ζητάει λοιπόν από τον παραλήπτη να τα μοιραστούν. Για να γίνει αυτό, ο παραλήπτης θα πρέπει να δώσει μερικά χρήματα (!)

1. Αυτό το τελευταίο δίνεται κι ως απάντηση σε παλαιά δήλωση γνωστού προσώπου, το οποίο δήλωνε πως «ό,τι είναι νόμιμο, είναι και ηθικό».

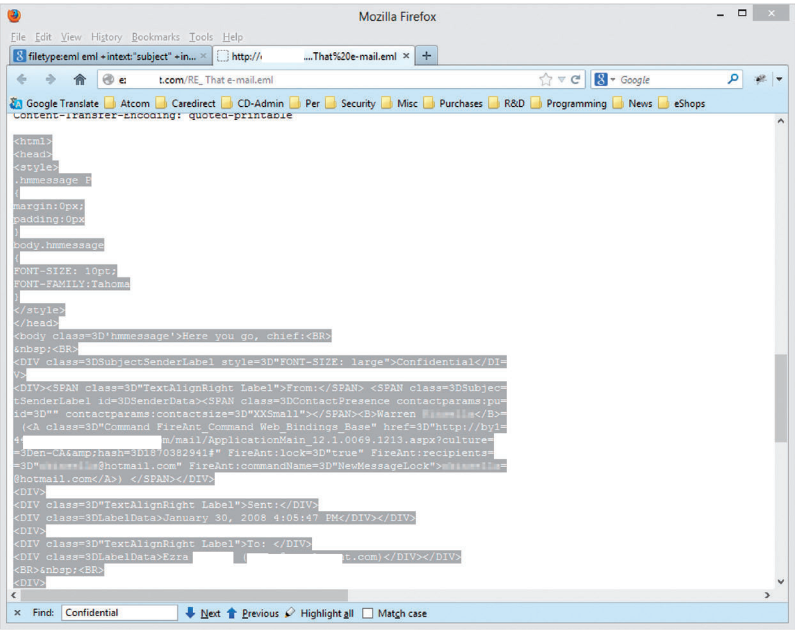


Εικόνα 10
Ότι είναι νόμιμο ΔΕΝ είναι κατά' ανάγκη και ηθικό... επιτέλους!



Εικόνα 11
Γεια σου φίλε. Βρήκα 970.000$. Θέλεις να τα φάμε μαζί;

ή τα credentials του λογαριασμού του, ώστε να γίνει η μεταφορά των χρημάτων. Όπως καταλαβαίνετε πρόκειται για μεγάλη απάτη! Ο κύριος Ahmed είναι ένας απατεώνας που περιμένει από τους ανυποψίαστους χρήστες να του στείλουν χρήματα, ή και τους κωδικούς των λογαριασμών τους, για να τους καταληστέψει! Ψάχνοντας περισσότερο πέσαμε πάνω σε ένα email γραμμένο σε HTML (εικόνα 12). Δεν είναι πολύ εύκολο να το διαβάσουμε, για αυτό και κάνουμε το εξής κόλπο: Το επιλέγουμε όλο και το κάνουμε Copy-Paste σε ένα αρχείο στο δίσκο μας και με την προέκταση html. Αφού το κάνουμε αυτό, μπορούμε να το ανοίξουμε και να το διαβάσουμε άνετα, αφού θα ανοίξει σε κάποιον browser (εικόνα 13).



Εικόνα 12
Αυτό το email είναι σε μορφή HTML. Θα το αποθηκεύσουμε ως ένα αρχείο HTML και θα το διαβάσουμε σαν κύριοι!

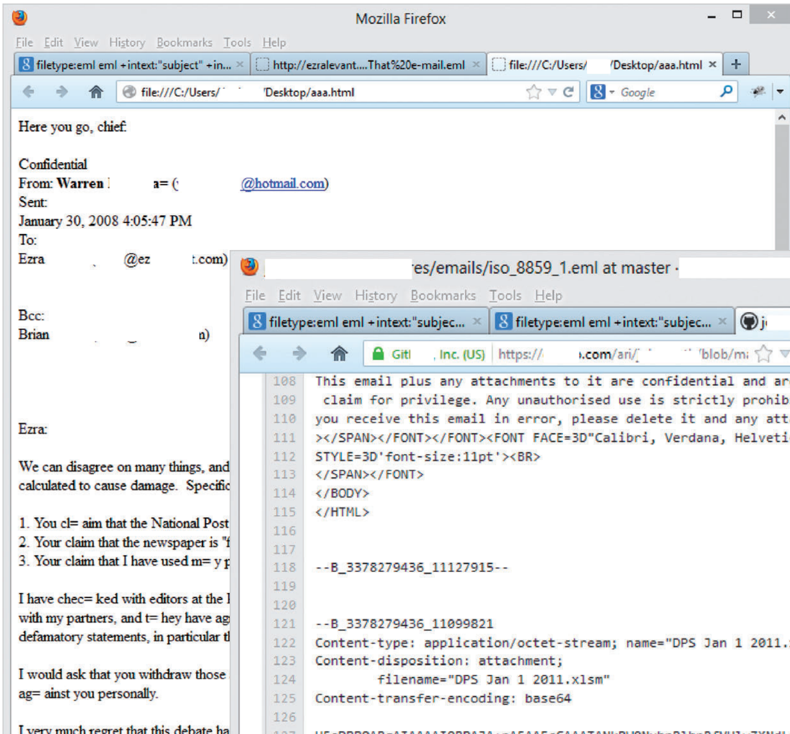
Το συγκεκριμένο email πάντως δεν περιέχει τίποτα εξαιρετικό. Πρόκειται για κάποιο blog που απειλείται με δικαστική αγωγή από κάποια εταιρεία, εξαιτίας κάποιων αναρτημένων σχολίων. Business as usual ;-) Ένα άλλο ενδιαφέρον εύρημα αποτελούν τα κρυπτογραφημένα emails και μάλιστα εκείνα που έχουν και υπογραφή (εικόνα 14). Αυτό σημαίνει οτι ο αποστολέας και ο παραλήπτης είναι απόλυτα πεπεισμένοι για την ασφάλεια της επικοινωνίας τους. Τι ειρωνεία!

Γιατί συμβαίνουν όλα αυτά;

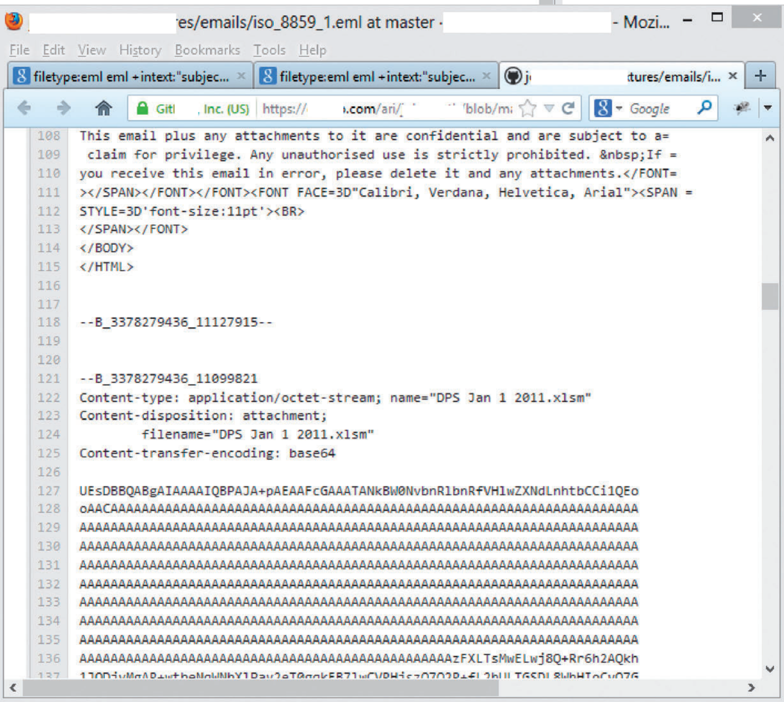
Μήπως φταίει το Google; Υπάρχουν άραγε αντίμετρα; Σκεφτείτε ότι βρισκόσαστε σ' έναν σκοτεινό χώρο, στον οποίο δεν βλέπετε τίποτα. Ευτυχώς, όμως, κρατάτε έναν φακό. Ανάβετε τον φακό και βλέπετε όλα τα προσωπικά πράγματα κάποιου, καθώς κι όλα του τα χρήματα επάνω σε ένα τραπέζι. Αν κλέβατε τα πράγματα και τα χρήματα, ποιος θα έφταιγε; Θα ήταν σωστό να πει κανείς ότι φταίει ο φακός; Μάλλον όχι! Θα μπορούσε μήπως να πει κανείς ότι φταίει εκείνος που άφησε τα υπάρχοντά του εκεί; Είναι δύσκολο να ισχυριστείς κάτι τέτοιο, διότι αυτός που τα άφησε πιθανόν να μη γνώριζε πόσο εύκολο ήταν να τα βρει κάποιος άλλος.

Το ζήτημα, λοιπόν, είναι αν θα καταφέρουμε ποτέ να είμαστε σίγουροι ότι τα προσωπικά μας δεδομένα είναι ασφαλή. Δυστυχώς, ποτέ δεν μπορείς να είσαι 100% σίγουρος για κάτι τέτοιο. Αυτό που μπορείς (και πρέπει) να κάνεις είναι να ενημερώνεσαι και να μαθαίνεις για το πώς λειτουργούν κάποια πράγματα, ώστε να καταλαβαίνεις τι μέτρα προστασίας χρειάζεται να πάρεις. Ας δούμε τρεις κανόνες που προκύπτουν από όσα είδαμε παραπάνω.

1. Δεν πρέπει να βάζουμε ποτέ τα backup των email μας (π.χ., τα αρχεία eml) σε καταλόγους ενός server, οι οποίοι είναι προσβάσιμοι από το web.



Εικόνα 13
Το email που βρήκαμε προηγουμένως, μέσα από τον browser μας...



Εικόνα 14
Ορίστε κι ένα κρυπτογραφημένο email! Η διαδικασία έπιασε τόπο ;)

2. Αν είμαστε αναγκασμένοι να κάνουμε κάτι τέτοιο, τότε τουλάχιστον ας τα βάλουμε κρυπτογραφημένα. Υπάρχουν πάρα πολλά, δωρεάν προγράμματα (π.χ. Palcrypt, http://en.wikipedia.org/wiki/Palcrypt, AxCrypt, www.axantum.com/axcrypt κ.ά.), που απλοποιούν ιδιαίτερα αυτή τη δουλειά.

3. Να μην επιτρέπουμε στην αράχνη της Google (αλλά και σε κάθε άλλη ανάλογη εταιρία/υπηρεσία) να μπαίνει σε συγκεκριμένους κατάλογους του server μας. Αυτό γίνεται πολύ εύκολα με τη δημιουργία και τη συντήρηση του περίφημου αρχείου robots.txt, http://en.wikipedia.org/wiki/Robots.txt. Πρόκειται για ένα αρχείο (το οποίο πρέπει να βρίσκεται στο λεγόμενο document-root του server) που περιέχει οδηγίες προς τις μηχανές αναζήτησης (http://bit.ly/crawlindex).

Ο γράφων πιστεύει ότι αν συνδυάσουμε τους παραπάνω τρεις κανόνες, είναι σχεδόν αδύνατον να μπορέσει να υποκλέψει κανείς τα προσωπικά μας δεδομένα. Εντάξει, τουλάχιστον όχι με τους τρόπους που είδαμε ;)

Η **διαφήμισή** σας θα μπορούσε να είναι εδώ.

Θα μπορούσε να 'ναι και σε κάποια άλλη σελίδα, δηλαδή.

Αλλά δεν είναι.

Προς το παρόν.

Πάντως μην αναρωτιέστε: Ξέρουμε ότι ξέρετε πως ξέρουμε ότι ξέρετε τι πρέπει να κάνετε.

Μην ντρέπεστε, λοιπόν.

talk2us@deltahacker.gr

Εμείς πάντως θα σας μιλήσουμε. (*)

(*) Στο μεταξύ συνεχίζουμε, **για δεύτερη συνεχή χρονιά,** με τη φροντίδα του καλύτερου τεχνολογικού περιοδικού γύρω από το ethical **hacking,** τα **δίκτυα** και την **ασφάλεια,** τον **προγραμματισμό** και τα **ηλεκτρονικά.**

Ξέρετε. Για **ανήσυχους, δημιουργικούς ανθρώπους** που δεν βλέπουν τον υπολογιστή τους ως τοστιέρα.

deltahacker.gr